Development of a Flight Control System for the Subsonic Wing Deployment of a Reusable Rocket Booster

by

Aaron Buysse

Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering in the Faculty of Engineering at Stellenbosch University

Department of Electrical and Electronic Engineering,

Stellenbosch University

Supervisor: Prof. WH Steyn

March 2018

UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

Plagiaatverklaring / *Plagiarism Declaration*

1. Plagiaat is die oorneem en gebruik van die idees, materiaal en ander intellektuele eiendom van ander persone asof dit jou eie werk is.
   Plagiarism is the use of ideas, material and other intellectual property of another's work and to present is as my own.

2. Ek erken dat die pleeg van plagiaat 'n strafbare oortreding is aangesien dit 'n vorm van diefstal is.
   I agree that plagiarism is a punishable offence because it constitutes theft.

3. Dienooreenkomstig is alle aanhalings en bydraes vanuit enige bron (ingesluit die internet) volledig verwys (erken). Ek erken dat die woordelikse aanhaal van teks sonder aanhalingstekens (selfs al word die bron volledig erken) plagiaat is.
   Accordingly all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism.

4. Ek verklaar dat die werk in hierdie skryfstuk vervat, behalwe waar anders aangedui, my eie oorspronklike werk is en dat ek dit nie vantevore in die geheel of gedeeltelik ingehandig het vir bepunting in hierdie module/werkstuk of 'n ander module/werkstuk nie.
   I declare that the work contained in this assignment, except where otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.

March 2018

Abstract

A flight control system and a navigation system have been developed for the subsonic wing deployment of a reusable first stage rocket booster. In previous research, the booster's wing was designed to rotate about a single pivot point, resulting in an oblique wing aircraft while partially deployed. Additionally, the booster was designed to utilise deployable propellers powered by a piston engine for thrust during the flyback portion of the return flight as an unmanned aircraft. This work defined a concept of operations for the booster for the subsonic return flight prior to starting the piston engine.

The aerodynamics of the booster were modelled in the rocket configuration, in the oblique wing configuration and in the aircraft configuration to form the basis for the controller designs. Furthermore, the critical Mach number of the booster's wing in the oblique wing configuration and in the aircraft configuration was determined, and constrains the airspeed at which the wing may be deployed. Since the critical Mach number increases with increasing wing sweep in the oblique wing configuration, the booster first partially deploys the wing to generate lift at a higher velocity and altitude. A pull-up manoeuvre was then performed to further reduce the airspeed by exchanging kinetic energy for gravitational potential energy, before fully deploying the wing. With the wing fully deployed, a minimum sink rate glide was established in order to maximise the time available to start the piston engine.

A cascaded control architecture was designed to handle the significant variation in the booster's geometry throughout the wing deployment, as well as the dynamic cross-coupling resulting from the oblique wing configuration. The control architecture consists of four parts. An outer loop uses single-input, single-output controllers to achieve the pull-up manoeuvre and the gliding flight by commanding an attitude. Two separate inner loops use multiple-input, multiple-output controllers to first track this attitude by commanding an angular velocity, and then track this angular velocity by commanding a body moment. Finally, control allocation is utilised to distribute the required moment to the six aerodynamic control surfaces on the booster, which vary in effectiveness and availability throughout the wing deployment.

The navigation system uses low-cost, commercially available sensors fused with two separate Kalman filters to estimate the attitude, position and velocity of the booster. A multiplicative extended Kalman filter estimates the booster's attitude using a gyroscope and measurements of inertially referenced vectors in the body frame. The reference vectors utilised were Earth's magnetic and gravity fields. The second Kalman filter estimates the position and velocity using the accelerometer and a Global Positioning System. The inertial velocity output from this filter was differentiated in order to estimate the dynamic acceleration, which was subtracted from an accelerometer measurement to provide the gravity vector measurement to the attitude filter.

The control system and navigation system were verified in a six degree-of-freedom simulation. Atmospheric disturbances were modelled by static winds, wind gusts and turbulence. Imperfect sensors were accounted for by introducing characteristic measurement noise in the simulation.

Uittreksel

'n Vlugbeheer- en navigasiestelsel is ontwikkel vir die subsoniese vlerkontplooing van 'n herbruikbare vuurpylstadium. In vorige navorsing is die vlerk van die eerste stadium ontwikkel om om 'n enkele spilpunt te kan roteer. Die resultaat hiervan is 'n skewe vlerk vliegtuig tydens vlerk ontplooing. Verder is die eerste stadium ontwerp om van ontplooibare propellers wat deur 'n suierenjin aangedryf word, gebruik te maak om stukrag te voorsien tydens die onbemande terugvlug. Hierdie werk definieer die konsep van die eerste stadium se werking tydens die subsoniese terugvlug, voordat die suierenjin aanskakel.

Die aerodinamiese eienskappe van die vuurpyl is gemodelleer in die vuurpylkonfigurasie, skeefvlerkkonfigurasie, sowel as die vliegtuigkonfigurasie om die basis van die beheerderontwerp te vorm. Verder is die kritiese Mach getal in beide die skeefvlerk- en vliegtuigkonfigurasie bepaal, aangesien dit die lugspoed waartydens die vlerk ontplooi kan word, beperk. Aangesien die kritiese Mach getal vermeerder soos die vlerkhoek vermeerder, ontplooi die eerstestadium die vlerk eers gedeeltelik sodat stygkrag teen 'n hoër spoed en hoogte gegenereer kan word. 'n Optrekbeweging word dan gedoen om die lugspoed verder te verminder deur kinetiese energie vir gravitasiepotensiële energie te verruil voordat die vlerk ten volle ontplooi word. Met die vlerk ten volle ontplooi, kan die eerste stadium teen 'n minimum dalingstempo sweef, sodat die tyd waarin die suierenjin kan aansit, gemaksimieer kan word.

'n Kaskade beheerargitektuur is ontwerp om die variasies in die vuurpylgeometrie tydens die vlerkontplooing sowel as die dinamiese kruiskoppeling in die skeef vlerk konfigurasie te hanteer. Die beheerargitektuur bestaan uit vier dele. 'n Buitelus-beheerder gebruik enkel-intree-enkel-uittree beheerders om beide die optrekbeweging sowel as die sweefvlug te beheer deur die oriëntasie te verander. Twee aparte binnelus-beheerders gebruik multi-intree-multi-uittree beheerders om eers die oriëntasie te volg deur die hoeksneheid te verstel en daarna hierdie snelheid te volg deur 'n ligaamsbeweging te maak. Uiteindelik word beheertoekenning gebruik om die nodige moment aan die ses beheervlakke toe te ken, aangesien die beheer oppervlakte se doeltreffendheid kan varieer tydens die vlerkontplooing.

Die navigasiestelsel gebruik lae koste, kommersieel-beskikbare sensors, geïntegreer met twee aparte Kalman-filters. Die filters word gebruik om die oriëntasie, posisie en snelheid van die vuurpyl te beraam. 'n Vermenigvuldigende, uitgebreide Kalman-filter skat die vuurpyl se oriëntasie deur van 'n giroskoop, sowel as metings van inersieel-verwysde vektore in die liggaam assestelsel gebruik te maak. Die verwysingsvektore gebruik die aarde se magneet- en swaartekragvelde. Die tweede Kalman-filter beraam die posisie en snelheid deur gebruik te maak van 'n versnellingsensor en GPS. Die inersiële snelheid uittree vanaf die filter word gedifferensieer om sodoende dinamiese versnelling te bepaal. Hierdie dinamiese versnelling word van die versnellingssensormetings afgetrek om die swaartekragvektore aan die oriëntasie filter te verskaf.

Beide die beheer- en navigasiestelsel is in 'n simulasie geverifieer waartydens ses vryheidsgrade in ag geneem is. Atmospheriese versteuringe is gemodelleer deur statiese winde, rukwinde sowel as turbulensie. Sensor meetfoute is nageboots deur die kenmerkende meetingsruis in die simulasie in te sluit.

Publications

Sections of the work presented in this thesis were published in the following conference paper:

Buysse, A., Steyn W.H. and Schutte, A., "Flight control system for a reusable rocket booster on the return flight through the atmosphere", *11th IAA Symposium on Small Satellites for Earth Observation,* Berlin, Germany, 24-28 April 2017.

Acknowledgements

I would like to thank the following people and organisations:

- Professor Herman Steyn for your vast knowledge, patience and advice throughout this research project. Your numerous insights have countlessly proven correct, and this research would not have been possible without you.
- Adriaan Schutte at Heliaq Advanced Engineering for constantly encouraging me to make the design more practical and for your advice throughout the project.
- Gabriël Roux for translating my abstract to Afrikaans.
- My mother, Sherri Buysse, for finding all my missing commas.
- My entire family for your support and encouragement throughout my studies.
- All my friends in the ESL for the many thought-provoking (although not always productive) conversations around the coffee machine.

In memory of my grandfather, Dale Buysse.

Contents

List of Figures

List of Tables

List of Acronyms

ADC Air Data Computer.

AHRS Attitude Heading Reference System.

ALV Austral Launch Vehicle.

CEP Circular Error Probability.

CFD Computational Fluid Dynamics.

DCM Direction Cosine Matrix.

DOF Degree of Freedom.

ECEF Earth Centered, Earth Fixed.

EKF Extended Kalman Filter.

ESL Electronic Systems Laboratory.

FADS Flush Air Data Sensing.

GPS Global Positioning System.

HWM Horizontal Wind Model.

IGRF International Geomagnetic Reference Field.

IMU Inertial Measurement Unit.

INS Inertial Navigation System.

LQR Linear Quadratic Regulator.

LQS Linear Quadratic Servo.

MEKF Multiplicative Extended Kalman Filter.

MEMS Microelectromechanical System.

MIMO Multiple-input, Multiple-output.

NED North-East-Down.

OWA Oblique Wing Aircraft.

OWRA Oblique Wing Research Aircraft.

PID Proportional-Integral-Derivative.

RLV Reusable Launch Vehicle.

SAS Stability Augmentation System.

SEP Spherical Error Probability.

SISO Single-input, Single-output.

SU2 Stanford University Unstructured.

UAV Unmanned Aerial Vehicle.

Nomenclature

Aerodynamics

ρ_∞ Air density

a Speed of sound

C_L, C_D Aerodynamic lift and drag coefficients

C_l, C_m, C_n Aerodynamic moment coefficients

C_P Aerodynamic pressure coefficient

C_X, C_Y, C_Z Aerodynamic force coefficients

M Mach number

n Load factor

Q_∞ Dynamic pressure

T_∞ Air temperature

U, V, W Airspeed vector in body coordinates

V_∞, α, β Polar airspeed vector

Constants

κ Ratio of specific heats for air (1.4)

g Gravitational acceleration ($9.81 m/s^2$)

R Ideal gas constant ($287 J/kg/K$)

Dynamics

χ Heading angle

γ Flight path angle

$\boldsymbol{\omega}$ Angular velocity vector

$\mathbf{A}$ Direction cosine matrix

$\mathbf{a}$ Acceleration vector

$\mathbf{f}$ Force vector

$\mathbf{h}$ Angular momentum vector

$\mathbf{m}$ Moment vector

$\mathbf{p}$ Linear momentum vector

$\mathbf{q}$ Quaternion

$\mathbf{r}$ Inertial position vector

$\mathbf{v}$ Inertial velocity vector

$\mathbf{v}$ Velocity vector

Φ, Θ, Ψ Roll, pitch and yaw angles

L, M, N Moment vector in body coordinates

N, E, D Inertial position vector in body coordinates

P, Q, R Angular velcity vector components in body coordinates

V_N, V_E, V_D Inertial velocity vector in body coordinates

X, Y, Z Force vector in body coordinates

Coordinate Frames

$\mathcal{A}$ Air Frame

$\mathcal{B}$ Body Frame

$\mathcal{E}$ Earth-Centred, Earth-Fixed Frame

$\mathcal{I}$ North-East-Down Frame

$\mathcal{W}$ Wing Frame

Controls

$\boldsymbol{\delta}$ Control input vector

A State coefficient matrix

B Input matrix

C Output matrix

K Control gain matrix

Q LQR state weighting matrix

R LQR input weighting matrix

r Reference vector

x State vector

a_1, a_2, c_1, c_2 Tuning parameters

K_i Integral gain

K_p Proportional gain

Estimation

ϵ Pitot-static probe bias

$\boldsymbol{\epsilon}_b$ Gyroscope bias

$\boldsymbol{\nu}$ Inertial reference vector

b Magnetic field vector

C Measurement output matrix

e Innovation vector

g Gravity field vector

L Estimator gain matrix

P Estimate covariance matrix

Q Process noise covariance matrix

R Measurement noise covariance matrix

y Measurement vector

Syntax

$\bar{x}$ Unit vector

$\hat{x}$ Estimated state

$\mathbf{X}$ Matrix

$\mathbf{x}$ Vector

x Variable

Vehicle Design

AR Aspect ratio

$\delta_{A,R}, \delta_{A,L}$ Right and left aileron

$\delta_{E,R}, \delta_{E,L}$ Right and left elevon

$\delta_{R,R}, \delta_{R,L}$ Right and left ruddervator

Λ Wing sweep angle

$\mathbf{I}$ Inertia tensor

b Span length

c Chord length

e Oswald efficiency factor

I_{xx}, I_{yy}, I_{zz} Moments of inertia

I_{xy}, I_{xz}, I_{yz} Products of inertia

m Mass

S Planform area

t Wing maximum thickness

Chapter 1

Introduction

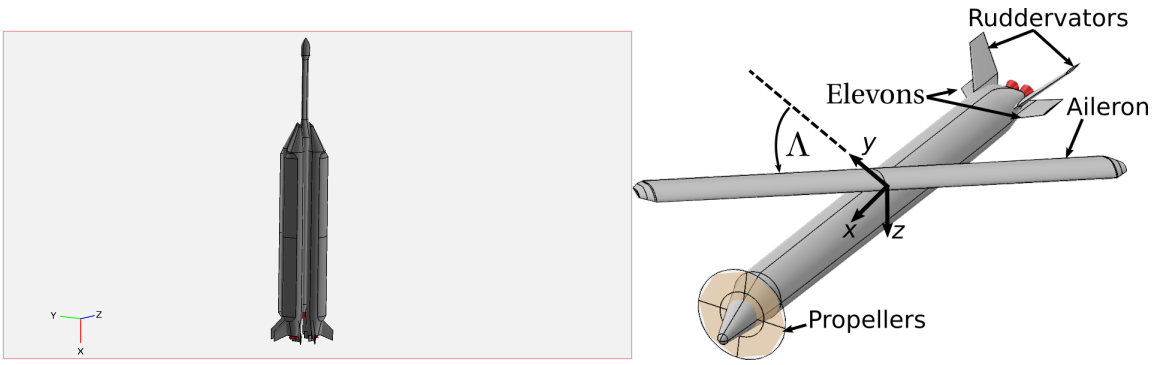
The rapid pace of technology advancements has significantly reduced the cost and size of satellites in recent years, allowing universities and small companies to design and manufacture highly-capable, small satellites. The cost of placing the satellite into orbit, however, severely limits this trend in cost reduction in part due to the practice of utilising expendable rocket boosters on the launch vehicle. In an effort to reduce this cost, the ride sharing practice has become common, in which small satellites, such as CubeSats, utilise leftover room in a launch vehicle carrying a larger satellite to orbit. Additionally, CubeSats are often included on crew resupply missions to and released from the International Space Station. These practices have helped to reduce launch costs, however they confine the smaller satellite's orbit and launch date to that of the largest payload.

Currently, there is no operational launch vehicle dedicated to small satellites, thus there is a need for a low cost launch vehicle to continue increasing accessibility to space for universities and small companies. Reusable launch vehicles have long been considered as a means to reduce costs and to increase launch frequency. This research supports the work being performed on one such vehicle, which is the Austral Launch Vehicle (ALV) being developed by Heliaq Advanced Engineering [1]. The ALV is a partially reusable, three-stage launch vehicle dedicated to placing small satellites into orbit with the goal of significantly reducing launch costs. This research aims to develop the flight control system for the crucial portion of the subsonic return flight of the ALV's reusable, first stage rocket boosters.

1.1 Austral Launch Vehicle

Figure 1.1 depicts the ALV in various configurations. The ALV consists of multiple, reusable first stage rocket boosters in a parallel launch configuration around expendable second and third stage rocket boosters. Each first stage booster has a deployable wing and deployable propellers powered by a piston engine, which are used to fly back as a large unmanned aerial vehicle (UAV). The first stage boosters' wing is initially stowed along the body longitudinal axis, and deploys by rotating about a single pivot point resulting in an oblique wing aircraft (OWA) while partially deployed. The propellers are stowed in the nose of the

booster and deploy by centrifugal force once the piston engine starts. The first stage boosters have six individually actuated control surfaces: two elevons and two ruddervators on the empennage and two ailerons on the wing.



(a) (b)

Figure 1.1: Austral Launch Vehicle: (a) launch configuration with four first stage boosters, (b) first stage booster with wing partially deployed.

Figure 1.2 shows the concept of operations for the ALV. The first stage boosters deliver the upper stages to the edge of the atmosphere, then separate and the expendable upper stages place the satellite into orbit. After separation, the first stage boosters enter an unpowered, ballistic flight to re-enter the atmosphere. They maintain a high angle of attack during hypersonic and supersonic re-entry in order to slow down. The first stage booster wing is then deployed once subsonic. The airspeed at which the wing may be deployed is limited by the critical Mach number, therefore the wing is first partially deployed to a non-zero sweep angle to increase the critical Mach number. Once the wing begins generating lift, the boosters pull up to exchange kinetic energy for potential energy in order to further slow down below the critical Mach number of the fully deployed wing. The piston engine then starts with the wings fully deployed at approximately 350 km downrange. The first stage boosters will then fly back to the launch pad or a runway, where they will be remotely landed by a human pilot. This research focuses on the development of the flight control system for the subsonic flight of a first stage booster throughout the deployment of the wing and up until the piston engine is started.

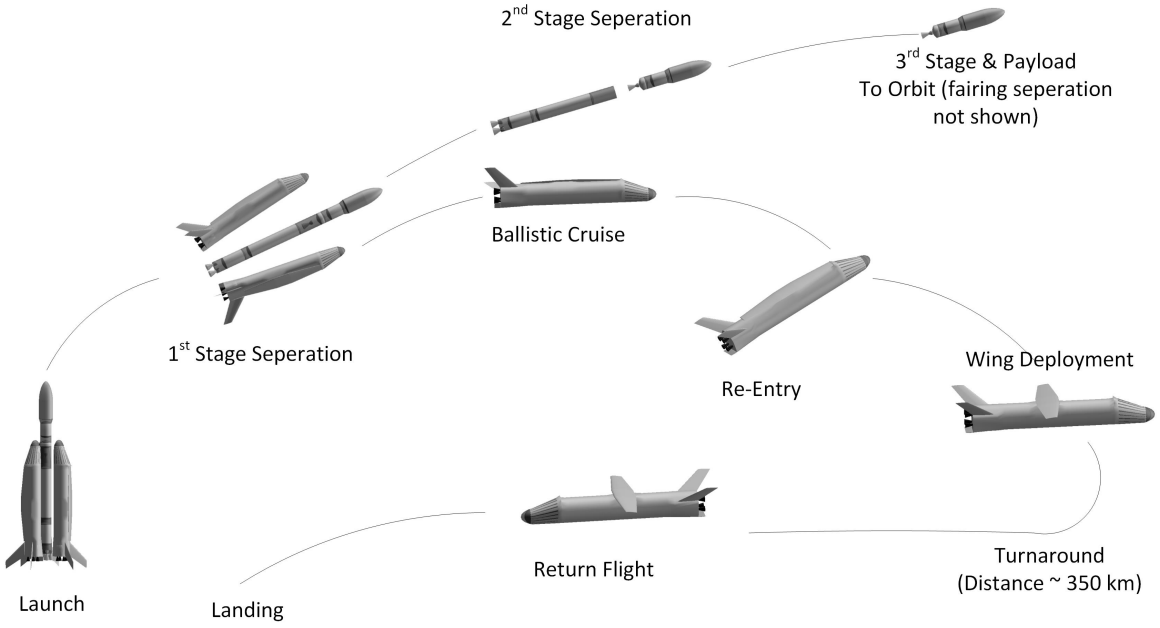


Figure 1.2: Austral Launch Vehicle Concept of Operations

1.2 Similar Aerospace Vehicles

While the ALV is entirely unique, the concepts utilised by the ALV have long been considered in the aerospace industry and provide an important foundation for the control system design. This section provides an overview of these vehicles, and a more detailed literature review of the flight control systems is provided in Chapter 2.

1.2.1 Reusable Launch Vehicles

Reusable launch vehicles (RLVs) have been proposed since the early days of the Space Race in order to help reduce costs and increase the frequency of launches. The United States' Space Shuttle was the first operational RLV, however development costs were extremely high and the frequency of launches was low. The Space Shuttle consisted of two Solid Rocket Boosters (SRBs) that were dropped into the ocean and refurbished after each flight, a reusable delta-wing orbiter that glided to a landing on a runway, and an expendable, external fuel tank. Efforts were made to make the SRBs fully reusable, as a significant amount of effort was required to locate and retrieve the SRBs from the ocean and to repair the corrosion caused by the salt water. Boeing performed a trade study [2] on various configurations for a Liquid Fly Back Booster (LFBB) to replace the SRBs. The three options considered were:

- Glide back: unpowered glide back to a runway,
- Boost-glide: use the rocket engines to regain energy and then glide back to a runway,
- Fly back: use aero engines to fly back to a runway.

The study traded mass, size and cost to determine that glide back was not feasible, while the fly back option required less mass and provided more operational flexibility over the boost-glide option. The study therefore selected the fly back option as the most feasible for a LFBB, and considered various aircraft configurations such as delta-wing, folding delta-wing, deployable wing rotating about a single pivot point (oblique wing) or two pivot points (scissor wing), and catamaran (dual fuselage). Although this study selected the catamarans as the baseline design, the oblique wing configuration was considered for the LFBB in other work [3].

Another option for RLVs that have been considered in the industry are powered return using the rocket boosters, with SpaceX's Falcon 9 first stage booster recently achieving operational status. Rocket-powered descent has minimal impact on the launch configuration, however requires a more sophisticated rocket engine that can be throttled and restarted in flight. More importantly, using the rocket engine to recover the booster reduces the short lifetime of the engine. Delta wings and catamarans have a large impact on the launch configuration, although have reduced complexity for the return flight due to the lack of moving parts. Deployable wings have a small impact on the launch configuration, but require additional mechanisms to deploy the wing during return. The oblique wing minimises these additional mechanism as compared to the scissor wings as only a single pivot point is required, however the asymmetrical properties during partial deployment introduces additional control challenges. Deployable wings have a further advantage in that the wing sweep can be optimised for the wide range of flight conditions that is encountered throughout the return flight. Furthermore, winged fly back vehicles provide increased operational flexibility as they are able to loiter near a runway before landing. The overall simplicity of the OWA fly back option and ability to draw upon the rich heritage of the aircraft industry motivated the selection of the OWA fly back configuration for the ALV first stage boosters.

1.2.2 Oblique Wing Aircraft

Extensive research has been performed on the aerodynamics and vehicle design of OWA [4] since they were first proposed for use as supersonic transport aircraft by R. T. Jones in 1958. However, OWA have yet to obtain operational maturity, in part due to the difficulty of controlling the asymmetrical vehicle. The primary challenge for the control of an OWA is that the wing inertially couples rolling and pitching motions, and aerodynamically couples motion about all three axes [5]. In the 1970s, NASA funded several studies to develop the necessary technology for an OWA. These culminated in the flight test of the manned AD-1 OWA, which was a low-speed technology demonstrator that flew with wing-sweep angles up to 60°. Significant reductions in handling qualities were observed above 45° due to cross coupling between pitching moments and aileron deflections, as well as due to poor roll stability [6]. However, the AD-1 was still controllable by a human pilot with sufficient experience and skill. The success of the AD-1 motivated NASA to begin development of a high-speed OWA, for which the NASA F-8 research aircraft was chosen to be modified. The majority of the research on flight control systems for OWA resulted from the F-8 program.

Thus, the current literature focuses on decoupling longitudinal and lateral motions so that a human pilot may fly an OWA as if it is a normal aircraft. As the ALV first stage boosters are completely autonomous, this explicit decoupling to pilot inputs is unnecessary and over-complicates the design process.

1.3 Research Objectives

A critical aspect of the success of the ALV is the capability to deploy the first stage boosters' wing. Thus, the primary objective of this research is to design a flight control system that stabilises the booster throughout the deployment of the wing. Once the wing has fully deployed, the aero piston engine will start, and the flight control system must provide as much time as possible to do this as the engine may not start on the first attempt.

Clearly, the significant challenge for this work is to design a control system capable of handling the multiple flight objectives (maximum time to start the aero engine and commence return flight), the continuously changing configuration as the wing deploys (from rocket to aircraft), and in such a way that minimises development costs (i.e. simple control system not dependent on high fidelity models and expensive sensors). Moreover, the ALV project is still in the conceptual design phase so it is highly desirable to have a modular and robust control system in order to ease the burden of design iteration. Specifically, this research aims to:

- Stabilise the attitude of the first stage booster in subsonic flight and throughout the deployment of the wing,
- Maximise the time available after deployment of the wing to start the piston engine,
- Determine the state of the booster using low-cost sensors.

1.4 Contributions

As flight control systems for traditional rockets and aircraft are well established in the literature, the primary focus and contribution of this research is the development of a flight control system that is capable of handling the transition from the rocket configuration to the aircraft configuration and the cross-coupling challenges introduced by the oblique wing configuration.

This work began by developing a low-fidelity aerodynamic modelling capability using open-source software to analyse the subsonic aerodynamics of the ALV first stage booster in the various configurations. This work flow provided a framework to quickly iterate on the design of the first stage booster while also providing the capability to model the interaction between the longitudinal and lateral dynamics that result from the oblique wing configuration.

Based on this model, additional control effectors, the elevons, were added to the booster empennage in order to make the booster fully controllable in the rocket configuration. The critical Mach number of the wing was determined at various wing sweep angles using computational fluid dynamics analysis and low-order theoretical analysis to determine the airspeed at which the wing could be deployed. A manoeuvre was then proposed to first partially deploy the wing, in order to deploy at a higher airspeed, and then pull up to exchange excess kinetic energy for potential energy before fully deploying the wing.

A novel cascade control architecture was developed to handle the cross-coupling in the oblique wing configuration and the significant change in the booster configuration. Single-input, single-output (SISO) outer-loop controllers meet the flight objectives using multiple-input, multiple-output (MIMO) controllers to control the attitude and angular velocity. Finally, state estimators were designed utilising a typical low-cost avionics system for an unmanned aerial vehicles to provide the states required for the flight control system.

In summary, the primary contributions of this research are:

- Development of an aerodynamic analysis work flow for the ALV first stage boosters,
- Addition of the elevons to achieve controllability in the rocket configuration,
- Development of a pull-up manoeuvre to deploy the wing at a higher airspeed and recover altitude,
- Development of a novel flight control architecture to handle the complex and varying vehicle,
- Development of state estimators using low-cost sensors.

1.5 Research Overview

This thesis is structured as follows:

- **Chapter 1. Introduction:** Introduces the Austral Launch Vehicle and how this research supports the development of a low-cost launch vehicle. The primary objectives and contributions this research makes to the ALV first stage booster flight control system are summarised.
- **Chapter 2. Literature Review:** Provides an overview of the current flight control systems for oblique wing aircraft as well as the state estimators commonly used on small unmanned aircraft. This chapter discusses the issues with the current flight control systems, and describes the control techniques that are utilised in order to address these issues.
- **Chapter 3. Vehicle Modelling:** Derives the equations of motion that describe the motion of the booster in three-dimensional space. This chapter first derives the full nonlinear kinematic and kinetic equations, and then describes the assumptions that are made in order to simplify the control system design process. This section describes the aerodynamic modelling of the booster, with a

focus on the unique properties of the ALV first stage booster. A potential flow software provides the aerodynamic coefficients as well as the stability and control derivatives that were derived in the equations of motion.

- **Chapter 4. Vehicle Design Analysis:** Analyses the equations of motion and unique properties of the ALV first stage booster. The unique control challenges posed by the ALV first stage boosters in the various configurations are discussed here. As a result of this analysis, additional control surfaces were added to the empennage in order to provide sufficient control in the rocket configuration. The limitations on the airspeed at which the wing is deployed are described, as well as the structural constraints on the pull-up manoeuvre. This analysis motivated the architecture and objectives of the flight control system described in the next chapter.

- **Chapter 5. Control System Design:** Develops the flight control architecture for the subsonic flight of the ALV first stage booster. The theoretical development for each loop of the control architecture is presented, along with descriptions and justifications of the design choices that resulted in the final flight control architecture. Outer-loop SISO controllers are presented to track a minimum sink-rate glide and a velocity heading angle using a reference attitude. A middle-loop MIMO controller is designed to track this reference attitude with zero steady-state error by commanding a reference angular velocity. An inner-loop MIMO controller tracks this reference angular velocity by commanding a moment. Finally, a control allocation algorithm distributes this moment to the physical control effectors in order to handle the redundant and varying availability of the control effectors. This control architecture requires several states that are difficult or impossible to measure directly, thus motivating the design of state estimators described in the next chapter.

- **Chapter 6. State Estimators Design:** Develops the state estimators for the ALV first stage booster. This chapter begins with a summary of the states required by the flight control system, describes the sensors considered and develops the estimation algorithms used to fuse data from the various sensors. A low-cost avionics suite is considered, consisting of an accelerometer, a gyroscope, a magnetometer, a GPS receiver and an air data measurement unit. Multiple Kalman filters are employed to combine the selected sensors in order to estimate the full state of the booster. The air data measurement unit has proven particularly challenging to integrate with the first stage boosters, and future consideration and impact on the control system design are discussed here.

- **Chapter 7. Simulation Results:** Presents the primary results of this research. The simulation environment used to verify the flight control system and state estimators is described. The design decisions discussed in the previous two chapters are analysed and compared.

- **Chapter 8. Conclusion:** Provides a summary of this research, discusses the overall contributions this work achieved as well as the shortcomings of the research. Future work to address these short-

comings and the work still required for the successful development of the ALV is recommended in this chapter.

Chapter 2

Literature Review

This chapter reviews the current literature on the design of flight control systems and navigation systems. The flight control system literature review focuses on flight control systems for oblique wing aircraft, as the first stage booster oblique wing configuration is the primary challenge of this research. The navigation systems are primarily required to obtain estimates of the booster's attitude, and the literature review focuses on attitude determination for aircraft.

2.1 Flight Control System

The primary challenge for the control system design of the ALV first stage boosters results from the oblique wing configuration. Furthermore, control systems for rockets and aircraft are very mature, and well described in [7]. Similarly, unmanned aircraft flight control is becoming a mature field and is well described in [8]. In these texts, the motion of an aircraft in inertial space may be modelled by 12 coupled, nonlinear ordinary differential equations. The states are the position of the vehicle, often represented in the local North-East-Down coordinate system, the velocity of the vehicle relative to the air, the inertial angular velocity of the vehicle and the inertial attitude of the vehicle, typically using the 3-2-1 Euler angle parameterisation. The north and east position and the yaw angle of the vehicle do not couple back into the equations of motion, and the down position only loosely couples into the equation of motion due to the variation in air density with altitude. Thus, the 12 equations of motion are typically reduced to eight equations of motion that, for symmetric vehicles, may be decoupled into four equations for the longitudinal dynamics and four equations for the lateral dynamics by assuming a small roll angle.

For OWA, this decoupling is not possible as the wing introduces inertial cross-coupling and aerodynamic cross-coupling [5]. The oblique nature of the wing results in a significant cross-product of inertia between the roll and pitch axes. Moreover, the swept forward portion of the wing experiences more downwash than the swept-back portion, resulting in a rolling moment to the side of the swept-forward portion of the wing. This increased downwash also results in more drag, causing a yaw moment that tends to unsweep the wing. Also, the increased longitudinal station of the ailerons will cause a significant pitching

moment at high sweep angles due to their offset from the centre of gravity, in addition to the reduced roll control effectiveness because of the shorter moment arm. As detailed in the introduction, OWA have primarily been considered for use as transport aircraft or military aircraft due to their ability to alter the wing sweep to obtain minimum drag across a wide variety of speeds. The flight control systems in the literature have therefore sought to decouple longitudinal and lateral motions so that a human pilot can fly the aircraft as if it is a symmetrical aircraft. Although these control systems are not directly applicable to the ALV, the design concepts and insight into the challenges of stabilising OWA is beneficial to the formulation of the control system design developed in this research. Therefore, the rest of this section reviews these stability augmentation systems.

2.1.1 Model Following Control

A significant amount of research has been performed for a stability augmentation system (SAS) for the NASA F-8 OWRA. Enns, *et al.* [9] used the MIMO loopshaping methodology in the frequency domain to provide a robust stability augmentation system that decoupled pilot stick and pedal inputs. The pitch coupling resulting from aileron deflection was avoided by using differential stabilisers on the empennage for roll control, while still using symmetric stabilisers for pitch control and the rudder for yaw control. Pilot evaluations of this stability augmentation system on the F-8 OWRA found that there were significant side forces in pitch maneuvers and that handling qualities deteriorated at the higher wing sweep angles [10].

Alag *et al.* [11] used model following control based on eigenstructure assignment to force the coupled eigenvectors of the linearised system for the F-8 OWRA at high wing sweep angles to follow the eigenvectors for the F-8 at zero wing sweep, thus decoupling the dynamics. This method considered the effectiveness of each control surface individually, and successfully decoupled the longitudinal and lateral dynamics. However, this method resulted in excessive control activity. Alag *et al.* [12] then used explicit model following control based on linear quadratic regulator (LQR) theory to regulate the error between the real state and a desired model state to zero. Again, the dynamics of the F-8 at zero wing sweep angle was used as the ideal model, and the effectiveness of each control surface was considered individually. This design resulted in less control activity, however steady state errors were large. Pahle [13] modified this method to use an output LQR in the model following framework to provide better insight into the design process, and added integrator states to reduce the steady state errors.

Morris [14] performed an in-depth analysis of the model following control methods and found that the closed loop dynamic modes are significantly faster than the models modes. This caused the excessive control activity seen in many of the previous works and implied that expensive, high speed actuators would be required. Furthermore, the large bandwidth resulting from these controllers is sensitive to the unmodelled, high frequency structural modes, such as aeroelasticity that is a common concern for oblique wing aircraft [15]. Morris then proposed a reduced order model following control method that selects gains by

numerically optimising the quadratic cost function, using the method developed by Ly [16]. This reduced order model following control provided much better model following capabilities for the transient and steady-state response, and eliminated the side force experienced during pitching maneuvers.

While these SAS developed using model following control successfully decouple the longitudinal and lateral motions, this approach is unnecessary for the ALV first stage boosters as the return flight will be fully autonomous and outer-loop controllers would still be required to replace the human pilot. However, several important trends are observed from these designs. First, MIMO control theory, in particular LQR design, has proven useful in handling the high cross-coupling and large number of independent actuators common in OWA. Furthermore, identifying a reduced order state for feedback is useful in order to gain intuition in the design and tuning of the controllers. Finally, it is important to manage the effectiveness of each individual control surface in the design of the controllers. This, in particular, is crucial for the first stage boosters, as the ailerons will be completely unavailable while the wing is stowed.

2.1.2 Nonlinear Dynamic Inversion

The SAS for the F-8 OWRA were all developed in the late 1980s, and were never flown since the program was terminated before an F-8 could be modified. Recently, there has been a renewed interest in OWA for high performance aircraft and near-space vehicles. Pang *et al.* [17] developed a flight control system using nonlinear dynamic inversion and the time-scale separation principle to divide the dynamics into several control loops. The inner-loop controlled the angular velocity, a middle-loop controlled the angle of attack, sideslip angle and roll angle, while outer-loops controlled the velocity vector in inertial space. The control surfaces management focused more on the thrust vectoring capabilities of the vehicle, and simply grouped the aerodynamic control surfaces into the typical aileron, elevator and rudder control surfaces. Lixin *et al.* [18] further expanded this method to use control allocation to consider the effectiveness of each individual control surface. The control allocation normalised the control variables by their maximum allowable deflection, and then used the pseudo-inverse of the control effectiveness matrix in order to use the smallest, normalised control power to achieve the desired inner-loop angular velocities.

Nonlinear dynamic inversion relies heavily on obtaining a very accurate model of the vehicle. Both of the previous two papers developed detailed dynamic models including all of the cross-coupling terms, however they did not consider the fidelity of the aerodynamic model used to obtain these terms. A major complication with using nonlinear dynamic inversion on the ALV first stage boosters results from the use of low fidelity aerodynamic modeling software. Furthermore, the vehicles considered in these works were not required to fully deploy the wing from a stowed position. The initial wing deployment for the first stage boosters is expected to result in highly nonlinear flow conditions that can only be accurately modeled by advanced computational fluid dynamic analysis or measured through wind tunnel or flight testing.

2.1.3 Proposed ALV Flight Control Architecture

The control architecture based on the time-scale separation principle from [17] and [18] is however a highly attractive approach, as it essentially separates the inertial cross-coupling and the aerodynamic cross-coupling into different control loops. The inner-loop tracks an angular velocity by commanding a moment, which handles the inertial cross-coupling. The middle-loop tracks an attitude relative to the airflow by commanding an angular velocity, thus handling the aerodynamic cross-coupling. This concept provided the basis for the control architecture developed in this research. Furthermore, the use of LQR theory from the F-8 OWRA program is embraced as an effective means of designing a robust MIMO controller, as well as the use of control allocation common throughout all of the literature on OWA as well as in the aerospace industry on advanced vehicles with control effector redundancies [19]. A high level block diagram of the flight control system proposed in this research is shown in Figure 2.1.

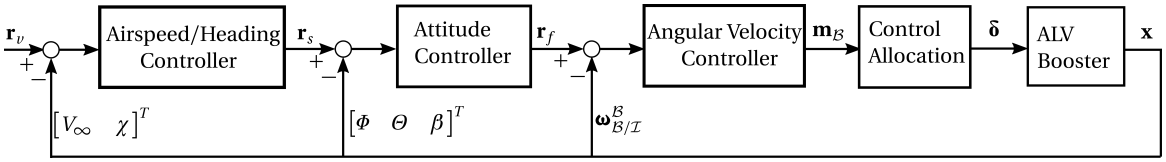


Figure 2.1: ALV First Stage Booster control architecture for the wing deployment.

In this proposed control architecture, a reference velocity state $\mathbf{r}_v$ is provided to the outer-loop controller in order to track a desired airspeed, V_∞, and a velocity heading angle, χ. The airspeed and heading angle are selected in order to meet the research objectives of providing maximum time to start the aero engine and beginning the return flight, and are discussed in more detail in Chapter 4. This outer-loop commands a slow state reference, $\mathbf{r}_s$, consisting of the roll angle Φ, pitch angle Θ and sideslip angle β. This middle-loop controller commands a reference angular velocity, $\mathbf{r}_f$ to the inner-loop controller, which commands a body moment $\mathbf{m}_\mathcal{B}$. Control allocation is then used to distribute this moment to the control effectors $\boldsymbol{\delta}$, while respecting the saturation limits and availability of the individual surfaces. Gain scheduling as a function of the Mach number and altitude is used to handle the wide flight envelope that will be encountered.

2.2 Navigation System

The states required for feedback in this work are the airspeed, heading angle, sideslip angle, roll angle, pitch angle and the angular velocity, and these states must be determined through the use of low-cost sensors. Furthermore, the angle of attack must be determined for the initial atmospheric re-entry of the first stage boosters and the position must be determined for navigation controllers not considered in this work. Kalman filters [20] are typically used in order to combine measurements from these sensors to provide estimates of the states, and is particularly important for the attitude states as these cannot be

measured directly. Since the Kalman filter is a linear estimator, variations of the Kalman filter, such as the Extended Kalman Filter (EKF) are typically used for attitude estimation.

2.2.1 Attitude Estimation

Euler angles are used for the controller design as they are intuitive to interpret and visualise. However, they are not ideal for attitude estimation since there is a singularity in the dynamics for some rotations and the dynamics are highly nonlinear because of the transcendental functions. Nonetheless, many works have developed state estimators for UAVs using Euler angles, such as those proposed by Eldrege [21] and Jung and Tsiotras [22]. A common theme with these estimators is the need to delegate estimation of the full aircraft state to several smaller Kalman filters in order to reduce computational complexity.

Further simplification in computational complexity can be obtained through the use of quaternions. Quaternions have the smallest dimension for a nonsingular attitude representation, and the dynamics are algebraic and nearly linear. However, they present some issues when used in an EKF. The four components of the quaternion are not independent since they satisfy the unity norm constraint, which results in a singularity of the covariance matrix in the EKF [23]. Markley [24] proposed a variation of the EKF, known as the Multiplicative EKF (MEKF), that makes use of a perturbation quaternion and quaternion multiplication for the measurement update of the quaternion estimate rather than the additive version for a standard EKF. The perturbation quaternion allows the MEKF to only estimate the vector portion of the quaternion, thereby reducing the number of attitude states in the Kalman filter from four to three.

The MEKF has its roots in spacecraft attitude estimators, for which there is a multitude of sensors for inertial vector measurements to choose from (i.e. sun sensors, star trackers, magnetometers or Earth horizon sensors). For air vehicles, the available inertial vectors are the magnetic field and the gravitational vector. Numerous works have adapted the MEKF to attitude estimation for air vehicles. In the ESL, Bijker developed an attitude heading reference system (AHRS) for an airship in which the attitude MEKF was used along with a separate EKF to estimate the position [25]. A crucial assumption in the use of multiple, small filters was that the airship was not accelerating, so the accelerometer measurement could be used as a direct measurement of gravity. The fly back of the ALV booster will experience periods of accelerating flight, so the dynamic acceleration must be removed from accelerometer measurements.

Kamali and Jain [26] developed an MEKF attitude estimator for an aircraft, however neglected the dynamic acceleration in the accelerometer measurements to sense gravity. Castellanos *et al.* [27] also used the MEKF for attitude estimation in a low-cost aircraft, and accounted for dynamic acceleration by assuming linear accelerations were only significant in the body longitudinal direction and thus could be calculated by differentiating an airspeed measurement. The accelerations due to turning was then computed from the cross product of the gyroscope measured angular velocity and the airspeed. However, Weibel *et al.* [28] showed that this common method of determining dynamic accelerations from airspeed and angular velocity measurements ignores lateral accelerations due to wind, and proposed differentiat-

ing the GPS measured velocity to estimate the dynamic acceleration. This will be important for the ALV first stage boosters as winds at various altitudes can be significant.

An additional complication for the ALV first stage boosters is that the delay on the GPS measurements will be significant due to the high velocity. Typical methods to account for delayed measurements in a Kalman filter are to propagate the states backwards in time to the time step corresponding to the delayed measurement, apply the measurement update, and then re-propagate the state forward to the current time step. This can be very computationally expensive, and so other methods proposed are to include additional states representing the delay. A very simple method proposed by Larsen *et al.* [29] uses the state estimate corresponding to the delay measurement in the measurement update of the time step that the measurement is received. This method does not guarantee optimality, however was shown in [22] to achieve good performance for a UAV.

All of the preceding research on the use of a MEKF for attitude estimation used the standard Kalman filter innovation term, in which the measurement is subtracted from the predicted state. Since attitude is not linear, it is desirable to make use of an innovation term that encapsulates the nonlinear nature of angular measurements. Martel *et al.* [30] proposed using a cross product of an inertial vector measured in body coordinates with the estimate of that vector as the innovation term in the MEKF for spacecraft. It is important to note that several researchers make use of a similar method, also based on [30], for attitude estimation of aircraft in the MEKF framework. Hall *et al.* [31] and Kim *et al.* [32] developed attitude estimators in which the full quaternion was propagated in the MEKF, and the perturbation quaternion was constructed from the cross product and dot product of vector measurements and their estimates. This however neglects one of the advantages of the MEKF, which allows the four dimensional quaternion to be reduced to a three dimensional vector for propagation in the EKF framework.

In summary, quaternions are highly desirable to use in attitude estimation since they are nonsingular and nearly linear. The MEKF further lowers the required computational complexity by reducing the dimension from the four component quaternion to the three components containing the vector portion, and the cross product innovation effectively accounts for the nonlinear nature of attitude. The two inertial reference vectors result from Earth's magnetic field and gravitational field, which are measured with a magnetometer and accelerometer. However, the dynamic acceleration of the vehicle must be accounted for to use the accelerometer for gravity vector measurements.

2.2.2 Proposed Attitude Heading Reference System

The proposed AHRS designed in this research combines various aspects from the previously described research. The task of estimating the position, velocity and attitude is divided into two separate Kalman filters. An MEKF is used for the attitude estimation, while using the cross production innovation to better describe the nonlinearity of attitude. A gyroscope is used to propagate a perturbation quaternion, and measurements of the magnetic field and gravity vector are utilised. The magnetic field is sensed directly

by a magnetometer, and the dynamic acceleration is subtracted from accelerometer measurements to sense the gravity vector. To account for all sources of dynamic acceleration, the inertial velocity vector is differentiated to obtain the dynamic acceleration. This inertial velocity is provided by a second Kalman filter propagated by the accelerometer and corrected with GPS measurements. The delay on the GPS measurement is handled in a computationally efficient manner by storing the state corresponding to the time step of the delayed measurement and updating at the time step this measurement is received.

2.2.3 Air Data Measurement Unit

The proposed AHRS provides estimates of the inertial position, velocity and attitude of the first stage boosters. An air data measurement unit is still required to obtain the velocity of the boosters relative to the air, which may differ from the inertial velocity due to static winds. Large transport aircraft typically use pitot-static probes and wind vanes mounted to the fuselage in order to measure the airspeed, angle of attack and sideslip angle. Small UAVs typically only use a pitot-static probe mounted to the wing for an airspeed measurement. If the angle of attack and sideslip angle are required for control, estimators, such as the one proposed by Johansen *et al.* [33], may be used to estimate the wind by comparing outputs of the AHRS with pitot-static probe measurements. With an estimate of the wind velocity and the vehicle's inertial velocity, the velocity relative to the air may be reconstructed and the flow angles computed with use of the attitude estimate.

For the ALV first stage boosters, sensors that protrude from the airframe are at risk of burning up during re-entry unless they are thermally protected or deployed once below hypersonic speeds. Both of these methods add significant mass and complexity, and therefore costs. Early flights of the Space Shuttle Orbiter deployed these sensors once below the hypersonic portion of re-entry, as atmospheric winds in the initial re-entry are typically not significant and therefore the flow angles can be determined from inertial measurements. In later flights, a Flush Air Data Sensing (FADS) system was used to obtain the flow angles [34]. The FADS system utilises a matrix of pressure ports integrated on the nose of a vehicle in order to measure the airspeed and flow angles. A FADS system is ideal for re-entry vehicles, as it has minimal impact on the vehicle design, can measure the flow angles across most of the flight envelope and is low cost. However, the design and calibration of a FADS is vehicle dependent and requires computational fluid dynamics, wind tunnel testing and flight testing to fully validate any proposed algorithm. Thus, the design and verification of a FADS system for the ALV is out of the scope of this work, although it is the recommended method for use on the ALV first stage boosters. This work will however evaluate the use of the wind estimator in [33] to obtain the airspeed and flow angles using measurements of the airspeed from a pitot-static probe and the output of the AHRS. Some of the insights obtained from this exercise may be extended to the FADS system design in future work.

Chapter 3

First Stage Booster Dynamic Model

This chapter derives the dynamic model of the booster, with the primary focus being to develop the equations of motion required for the control system design. The first stage booster is shown in Figure 3.1. The booster has six independently actuated control surfaces: right aileron $\delta_{A,R}$, left aileron $\delta_{A,L}$, right elevon $\delta_{E,R}$, left elevon $\delta_{E,L}$, right ruddervator $\delta_{R,R}$ and left ruddervator $\delta_{R,L}$. The ailerons' chord length is 25% of the wing chord, and the elevons' and ruddervators' chord length are the entire tail fin chords. Throughout this thesis, a positive control surface deflection is downwards towards the positive body z-axis. The wing sweep angle Λ is measured from the body y-axis.

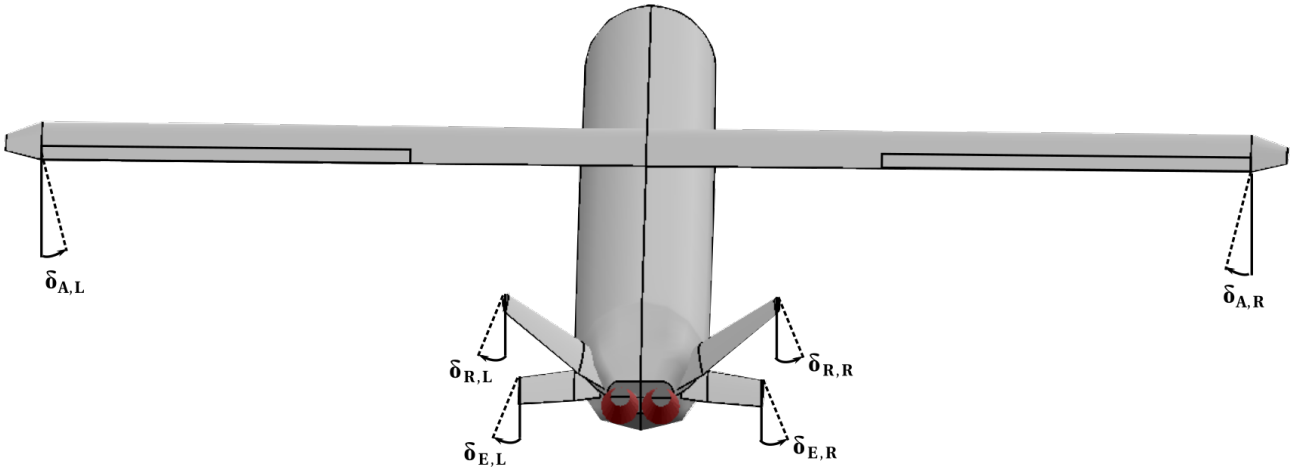


Figure 3.1: ALV first stage booster independently actuated control surfaces. There are two ailerons on the wing,

3.1 Coordinate Frames

This section introduces the coordinate frames used to model the motion of the booster in inertial space as well as to model the deployment of the wing.

3.1.1 Inertial Frame

Space vehicles are often modeled in the Earth-Centered Earth-Fixed (ECEF) reference frame, however for the subsonic return flight a local reference frame fixed to the surface of Earth may be used. The North-East-Down (NED) reference frame is commonly used for aircraft, which assumes a flat, non-rotating Earth. In the NED reference frame, the origin is fixed to a convenient location on the surface of Earth, such as the launchpad. The $\bar{x}_{\mathcal{I}}$ axis points towards true north, the $\bar{y}_{\mathcal{I}}$ axis points true east, and the $\bar{z}_{\mathcal{I}}$ axis completes the right-handed coordinate frame pointing downwards. Figure 3.2 shows the ECEF and NED coordinate frames.

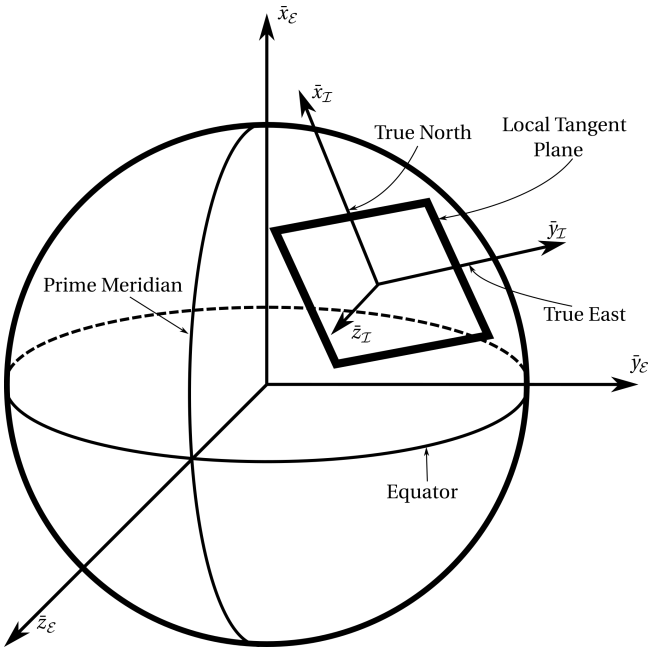


Figure 3.2: ECEF and NED Coordinate Frames

3.1.2 Body Frame

A noninertial frame fixed to the booster body is used to describe the position in inertial space. The origin is located at the centre of gravity, the $\bar{x}_{\mathcal{B}}$ axis points out the nose, the $\bar{y}_{\mathcal{B}}$ axis points out the right wing at zero sweep, and the $\bar{z}_{\mathcal{B}}$ axis completes the right-handed coordinate system pointing through the bottom of the booster. Figure 3.3 shows the body coordinate frame referenced to the NED frame. In this work, a superscript frame notation denotes the coordinate frame in which a vector is represented, and the velocity of one frame relative to another is denoted in a subscript. For example, $\boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}$ is the angular velocity of the body frame relative to the inertial frame, represented in body frame coordinates.

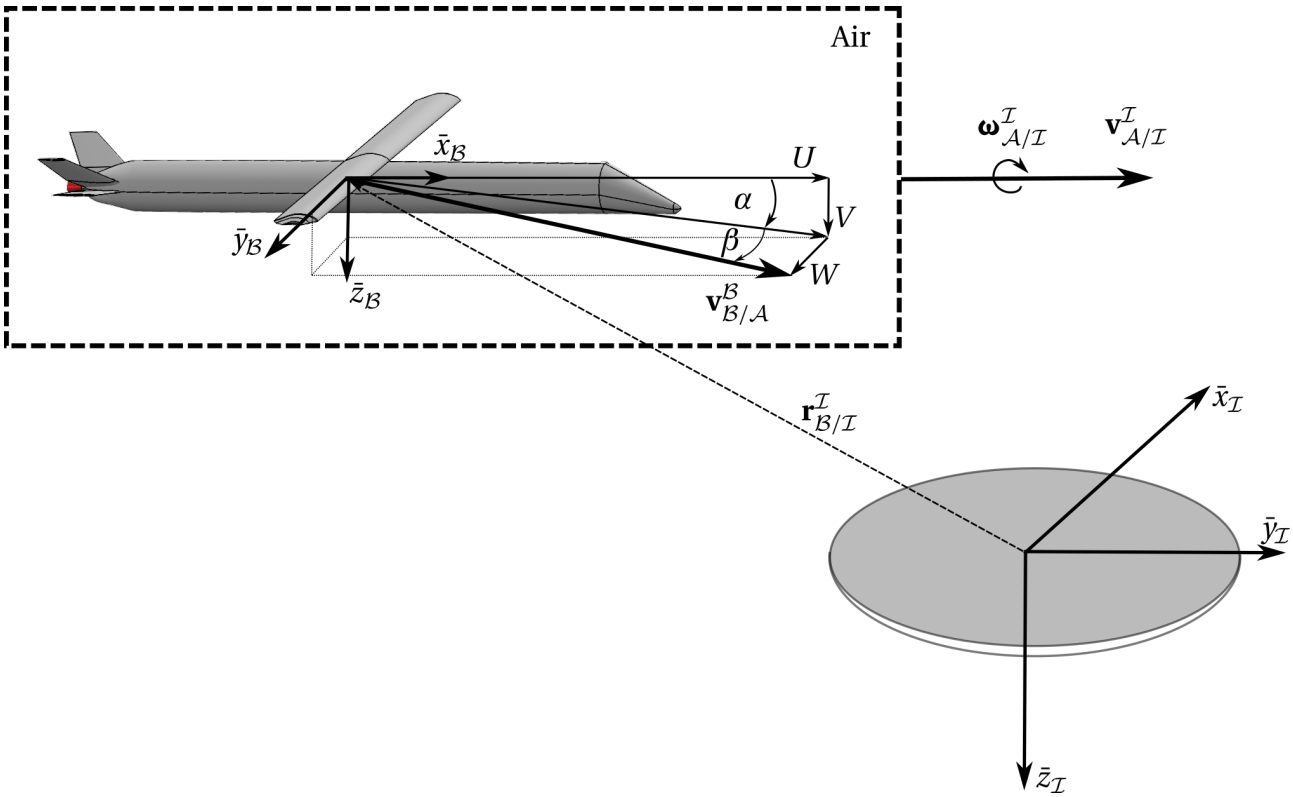


Figure 3.3: NED and Body Coordinate Frames

3.1.3 Stability Frame

The lift and drag on the booster result from the motion of the booster relative to the air, and are modelled in the stability frame. The $\bar{x}_{\mathcal{S}}$ axis is the projection of the velocity vector of the booster relative to the air $\mathbf{v}_{\mathcal{B}/\mathcal{A}}$, referred to as the airspeed vector, onto the body $x-z$ plane, the $\bar{y}_{\mathcal{S}}$ axis remains aligned with the $\bar{y}_{\mathcal{B}}$ axis, and the $\bar{z}_{\mathcal{S}}$ axis completes the right-handed orthogonal set. The stability frame is fixed to the booster's centre of gravity.

3.1.4 Wing Frame

A wing frame is defined to model the rotation of the wing relative to the booster. The wing frame is fixed to the centre of gravity of the wing, with the $\bar{x}_{\mathcal{W}}$ axis pointing forward along the wing chord, the $\bar{y}_{\mathcal{W}}$ axis pointing to the right side of the wing along the wing span, and the $\bar{z}_{\mathcal{W}}$ axis completing the right-handed orthogonal set. Figure 3.4 shows the wing frame relative to the body frame with a non-zero wing sweep.

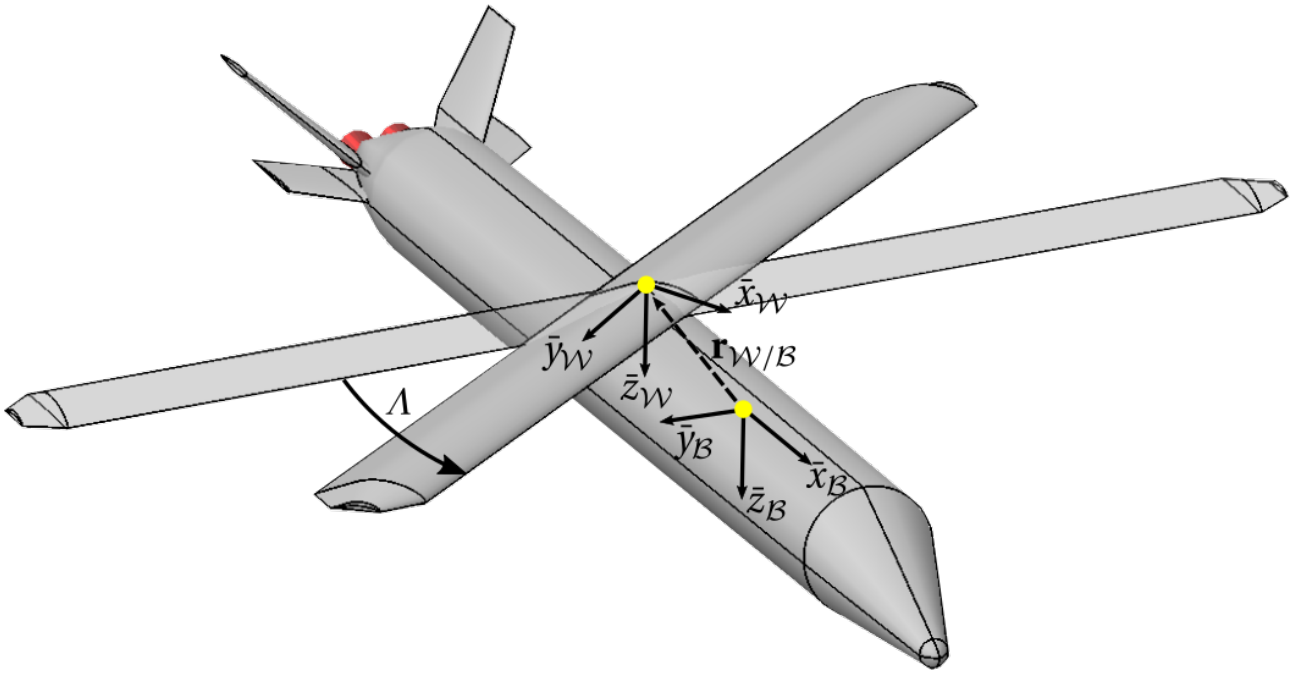


Figure 3.4: The booster wing and body frames. Note that the position of the booster centre of gravity is exaggerated to aid visualisation. The booster centre of gravity is located approximately a quarter of the wing chord behind the wing leading edge, under the wing centre of gravity.

3.1.5 Attitude Representation

The unit vectors of a given frame may be rotated to another frame by a Direction Cosine Matrix (DCM), **A**. The DCMs for single-axis rotations about each of the three unit vectors are

$$\mathbf{A_1}(\theta)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos\theta & \sin\theta\\ 0 & -\sin\theta & \cos\theta\end{bmatrix},\quad \mathbf{A_2}(\theta)=\begin{bmatrix}\cos\theta & 0 & -\sin\theta\\ 0 & 1 & \\ \sin\theta & 0 & \cos\theta\end{bmatrix},\quad \mathbf{A_3}(\theta)=\begin{bmatrix}\cos\theta & \sin\theta & 0\\ -\sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}. \tag{3.1.1}$$

3.1.6 Inertial to Body Frame

For aircraft, the attitude of the body frame relative to the inertial frame is parameterised by the 3-2-1 Euler angle set, with yaw angle Ψ, pitch angle Θ, and roll angle Φ as shown in Figure 3.5. The inertial frame unit vectors are first rotated about the $\bar{z}$ axis by Ψ, then about the resulting $\bar{y}$ axis by Θ, and finally about the $\bar{x}$ axis by Φ resulting in the body unit axes.

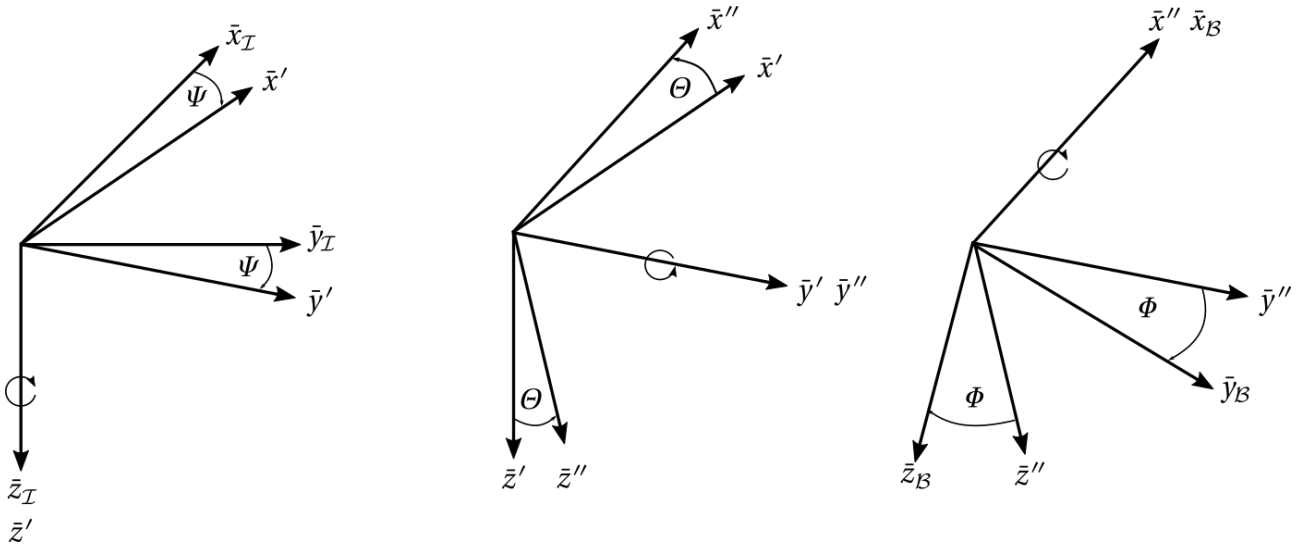


Figure 3.5: Euler angle 3-2-1 set from inertial to body coordinates.

The DCM is constructed as

$$\begin{aligned}\mathbf{A}_{\mathcal{I}}^{\mathcal{B}} &= \mathbf{A_1}(\Phi)\mathbf{A_2}(\Theta)\mathbf{A_3}(\Psi)\\ &= \begin{bmatrix} \cos\Theta\cos\Psi & \cos\Theta\sin\Psi & -\sin\Theta \\ \sin\Phi\sin\Theta\cos\Psi-\cos\Phi\sin\Phi & \sin\Phi\sin\Theta\sin\Psi+\cos\Phi\cos\Phi & \sin\Phi\cos\Theta \\ \cos\Phi\sin\Theta\cos\Psi+\cos\Phi\cos\Phi & \cos\Phi\sin\Theta\sin\Psi-\sin\Phi\cos\Phi & \cos\Phi\cos\Theta \end{bmatrix}.\end{aligned} \tag{3.1.2}$$

Thus, the Euler angles may be obtained from the DCM as

$$\begin{aligned}\Phi &= \text{atan2}\left(\frac{A_{23}}{A_{33}}\right),\\ \Theta &= -\sin^{-1}(A_{13}),\\ \Psi &= \text{atan2}\left(\frac{A_{12}}{A_{11}}\right),\end{aligned} \tag{3.1.3}$$

where atan2() is the four-quadrant inverse tangent function. The 3-2-1 Euler angle parameterisation is used in the design of the control system as they are intuitive to interpret and visualise. However, they are not ideal for attitude estimation as the dynamics are highly nonlinear and contain singularities for some rotations. Alternatively, quaternions can also be used to parameterise the attitude of the body frame. Quaternions are a nonsingular attitude representation defined in terms of Euler's rotation theorem, which states the attitude may be described by a rotation θ about an arbitrary axis $\mathbf{u} = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^T$. The quater-

nion is defined as

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} u_1 \sin\left(\frac{\theta}{2}\right) \\ u_2 \sin\left(\frac{\theta}{2}\right) \\ u_3 \sin\left(\frac{\theta}{2}\right) \\ \cos\left(\frac{\theta}{2}\right) \end{bmatrix} = \begin{bmatrix} \mathbf{q}_{vec} \\ q_4 \end{bmatrix}. \tag{3.1.4}$$

The DCM in terms of the quaternion is

$$\mathbf{A}_{\mathcal{I}}^{\mathcal{B}} = \begin{bmatrix} q_4^2 + q_1^2 - q_2^2 - q_3^2 & 2\left(q_4 q_3 + q_1 q_2\right) & 2\left(q_1 q_3 - q_4 q_2\right) \\ 2\left(q_1 q_2 - q_4 q_3\right) & q_4^2 - q_1^2 + q_2^2 - q_3^2 & 2\left(q_1 q_4 + q_2 q_3\right) \\ 2\left(q_4 q_2 + q_1 q_3\right) & 2\left(q_2 q_3 - q_4 q_1\right) & q_4^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}. \tag{3.1.5}$$

3.1.7 Body to Stability Frame

The stability frame results from left-handed rotation about the $\bar{y}_{\mathcal{B}}$ axis by the angle of attack, α. Thus, the DCM for rotations from the body to the stability frame is

$$\mathbf{A}_{\mathcal{B}}^{\mathcal{S}} = \mathbf{A_2}(-\alpha). \tag{3.1.6}$$

3.1.8 Body to Wing Frame

The wing frame results from a rotation about the $\bar{z}_{\mathcal{B}}$ axis by the wing sweep angle, Λ. Thus, the DCM for rotations from the body to the wing frame is is

$$\mathbf{A}_{\mathcal{B}}^{\mathcal{W}} = \mathbf{A_3}(\Lambda). \tag{3.1.7}$$

3.2 Kinematics

This section derives the kinematics of the booster relating the linear and angular velocity to the position and attitude, respectively.

3.2.1 Linear Position and Velocity

As seen in Figure 3.3, the position of the body frame in the inertial frame, $\mathbf{r}_{\mathcal{B}/\mathcal{I}}$ is expressed in inertial frame coordinates as

$$\mathbf{r}_{\mathcal{B}/\mathcal{I}}^{\mathcal{I}} = \begin{bmatrix} N \\ E \\ D \end{bmatrix}. \tag{3.2.1}$$

Figure 3.6 shows the inertial velocity vector, $\mathbf{v}_{\mathcal{B}/\mathcal{I}}$ in the inertial frames. The velocity vector expressed in inertial frame coordinates is

$$\mathbf{v}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I}} = \dot{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I}} = \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix}. \tag{3.2.2}$$

The inertial velocity vector can be defined in terms of the heading angle, χ, as

$$\chi = \tan^{-1}\left(\frac{V_E}{V_N}\right), \tag{3.2.3}$$

and the flight path angle, γ, as

$$\gamma = \tan^{-1}\left(\frac{-V_D}{\sqrt{V_N^2 + V_E^2}}\right). \tag{3.2.4}$$

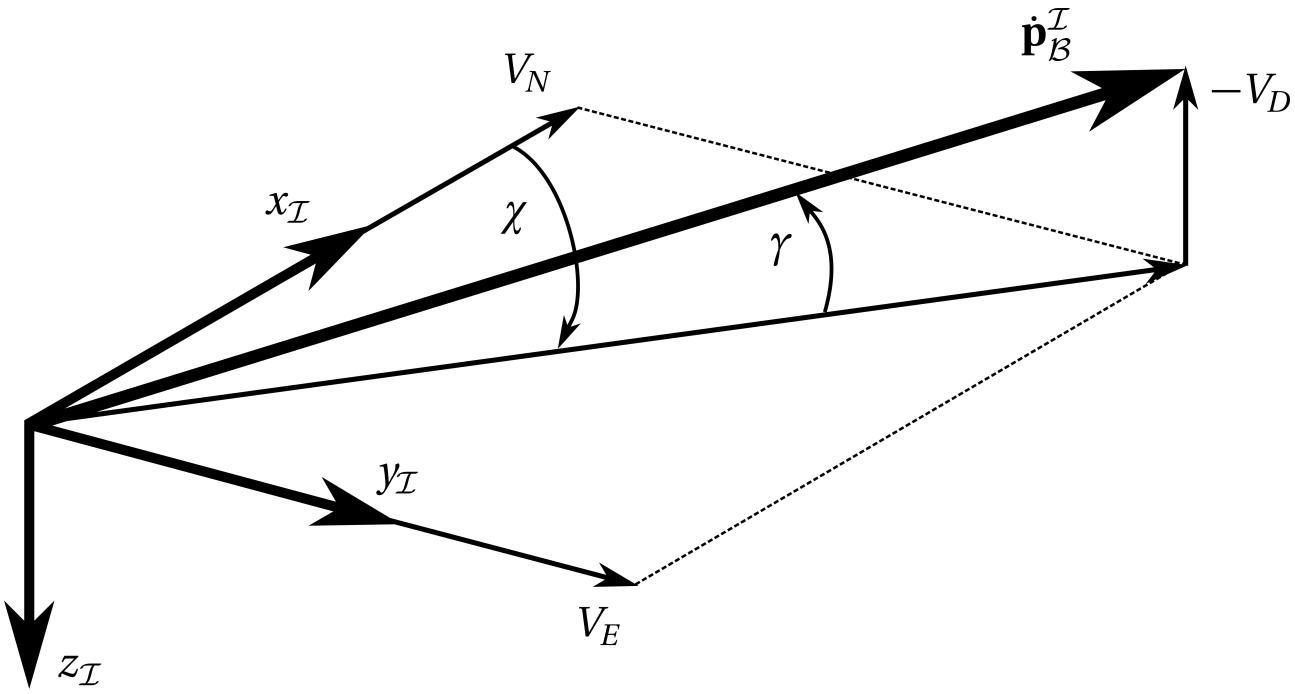


Figure 3.6: Velocity vector in the NED frame.

However, the aerodynamic forces and moments result from the velocity of the booster relative to the air, described by the airspeed vector expressed in body coordinates as

$$\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A}} = \begin{bmatrix} U \\ V \\ W \end{bmatrix}, \tag{3.2.5}$$

and may differ from the inertial velocity if there is a nonzero velocity of the air relative to the inertial frame,

$\mathbf{v}_{\mathcal{A}/\mathcal{I}}$, resulting from winds. The relationship between the inertial velocity vector and airspeed vector is

$$\mathbf{v}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I}} = \mathbf{A}^{\mathcal{I}}_{\mathcal{B}}\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A}} + \mathbf{v}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I}} = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}\right]^T \mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A}} + \mathbf{v}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I}}. \tag{3.2.6}$$

The airspeed vector is commonly expressed in polar format in terms of the airspeed, V_∞, the angle of attack, α, and the sideslip angle, β, defined as

$$V_\infty = \sqrt{U^2 + V^2 + W^2}, \tag{3.2.7}$$

$$\alpha = \tan^{-1}\left(\frac{W}{U}\right), \tag{3.2.8}$$

and

$$\beta = \sin^{-1}\left(\frac{V}{V_\infty}\right). \tag{3.2.9}$$

Using the polar representation, the airspeed vector expressed in body frame coordinates is

$$\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A}} = V_\infty \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\beta \\ \sin\alpha\cos\beta \end{bmatrix}. \tag{3.2.10}$$

3.2.2 Angular Velocity

The angular velocity of the body frame relative to the inertial frame, $\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}$, is expressed in body frame coordinates as

$$\boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} = \begin{bmatrix} P \\ Q \\ R \end{bmatrix}. \tag{3.2.11}$$

The rate of change of the Euler angles differs from the angular velocity because of the intermediate rotations. The relationship is given by

$$\begin{aligned}\begin{bmatrix} P \\ Q \\ R \end{bmatrix} &= \begin{bmatrix} \dot{\Phi} \\ 0 \\ 0 \end{bmatrix} + \mathbf{A_1}(\Phi) \begin{bmatrix} 0 \\ \dot{\Theta} \\ 0 \end{bmatrix} + \mathbf{A_1}(\Phi)\mathbf{A_2}(\Theta) \begin{bmatrix} 0 \\ 0 \\ \dot{\Psi} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & -\sin\Theta \\ 0 & \cos\Phi & \sin\Phi\cos\Theta \\ 0 & -\sin\Phi & \cos\Phi\cos\Theta \end{bmatrix} \begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix}.\end{aligned} \tag{3.2.12}$$

Inverting the matrix provides the dynamics of the 3-2-1 Euler angle parameterisation as

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \sin\Phi\sec\Theta & \cos\Phi\sec\Theta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix}. \tag{3.2.13}$$

Thus, there is a singularity in the Euler angle dynamics for $\Theta = \pm 90°$. Although the booster should remain far from this singularity during the return flight, it will be launched vertically and thus motivated the use of quaternions in addition to Euler angles. The quaternion rates for a body angular velocity are given as

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_4 & -q_3 & q_2 & q_1 \\ q_3 & q_4 & -q_1 & q_2 \\ -q_2 & q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \\ 0 \end{bmatrix}. \tag{3.2.14}$$

The quaternion dynamics are linear, only require algebraic computations, and avoid the singularity encountered with Euler angles.

3.3 Kinetics

This section develops the relations between the external forces and moments and the result on the linear and angular velocity of the booster.

3.3.1 Linear Acceleration

From Newton's second law, the sum of all forces acting on the booster, $\mathbf{f}_b$, equals the rate of change of the linear momentum, $\mathbf{p}_b$, in the inertial frame

$$\mathbf{f}_b = {}^{\mathcal{I}}\dot{\mathbf{p}}_b = \frac{d}{dt}{}^{\mathcal{I}}\left[m\dot{\mathbf{r}}_{\mathcal{B}/\mathcal{I}}\right], \tag{3.3.1}$$

where m is the mass of the booster. It is convenient to work with the forces represented in the body frame. The transport theorem may be used to express the inertial derivative in Equation 3.3.1 in terms of the derivative in the body frame. Prior to starting the piston engine, the mass constant, and the inertial derivative of the velocity may be represented in the body frame as

$$\mathbf{f}_b = m\dot{\mathbf{v}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} + \boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} \times \left(m\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}\right). \tag{3.3.2}$$

Equation 3.3.2 can be rearranged in terms of the linear acceleration as

$$\dot{\mathbf{v}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} = \frac{1}{m}\mathbf{f}_{\mathcal{B}} - \left[\boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}\times\right]\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}, \tag{3.3.3}$$

where $\left[\boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}\times\right]$ is the skew-symmetric matrix representing the cross product operator.

3.3.2 Angular Acceleration

Similarly, the sum of all moments acting on the booster, $\mathbf{m}_{\mathcal{B}}$, equals the rate of change of the angular momentum, $\mathbf{h}$, in the inertial frame

$$\mathbf{m}_b = {}^{\mathcal{I}}\dot{\mathbf{h}}_b = \frac{d}{dt}{}^{\mathcal{I}}\left[\mathbf{I}_b\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}\right], \tag{3.3.4}$$

where the inertia tensor, $\mathbf{I}_b$, is defined as

$$\mathbf{I}_b = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}. \tag{3.3.5}$$

The inertia tensor of the booster varies as the wing deploys, thus the inertia is considered in terms of the contributions from the wing, $\mathbf{I}_w$, and from the fuselage $\mathbf{I}_f$. Thus,

$$\mathbf{I}_b = \mathbf{I}_w + \mathbf{I}_f = \mathbf{I}_{w,0} + \tilde{\mathbf{R}}_w + \mathbf{I}_{f,0} + \tilde{\mathbf{R}}_f, \tag{3.3.6}$$

where $\mathbf{I}_{w,0}$ and $\mathbf{I}_{f,0}$ are the inertia tensors of the wing and fuselage, respectively, about their own centre of gravity and $\tilde{\mathbf{R}}_w$ and $\tilde{\mathbf{R}}_f$ are the terms from the parallel axis theorem due to the offset of the wing and fuselage from the booster's centre of gravity. These are given by

$$\tilde{\mathbf{R}} = m\begin{bmatrix} y_G^2 + z_G^2 & x_G y_G & x_G z_G \\ x_G y_G & x_G^2 + z_G^2 & y_G z_G \\ x_G z_G & y_G z_G & x_G^2 + y_G^2 \end{bmatrix}, \tag{3.3.7}$$

where $\left[x_G, y_G, z_G\right]^T$ is the vector from the booster centre of gravity to the wing or fuselage centre of gravity. During the wing deployment, it is assumed that the centre of gravity of the wing remains constant relative to the vehicle centre of gravity. It is also assumed that the wing rotates at a constant angular velocity of

$$\boldsymbol{\omega}_{w/b}^{\mathcal{B}} = \begin{bmatrix} 0 \\ 0 \\ \dot{\Lambda} \end{bmatrix} \tag{3.3.8}$$

during the wing deployment. Thus, the angular momentum of the booster is

$$\mathbf{h}_b = \mathbf{I}_b \boldsymbol{\omega}_{b/i} + \mathbf{I}_w \boldsymbol{\omega}_{w/b}. \tag{3.3.9}$$

Substituting Equation 3.3.9 into Equation 3.3.4 gives

$$\mathbf{m}_b = \frac{d}{dt}^{\mathcal{I}} \left[\mathbf{I}_b \boldsymbol{\omega}_{b/i} + \mathbf{I}_w \boldsymbol{\omega}_{w/b} \right], \tag{3.3.10}$$

which is expanded to

$$\mathbf{m}_b = \frac{d}{dt}^{\mathcal{B}} \left[\mathbf{I}_b \boldsymbol{\omega}_{b/i} + \mathbf{I}_w \boldsymbol{\omega}_{w/b} \right] + \boldsymbol{\omega}_{b/i} \times \left[\mathbf{I}_b \boldsymbol{\omega}_{b/i} + \mathbf{I}_w \boldsymbol{\omega}_{w/b} \right] \tag{3.3.11}$$

using the transport theorem. Since, the angular momentum of the wing in the body frame is assumed constant with respect to the body frame during the wing deployment, Equation 3.3.11 simplifies to

$$\mathbf{m}_b = \dot{\mathbf{I}}_b \boldsymbol{\omega}_{b/i} + \mathbf{I}_b \dot{\boldsymbol{\omega}}_{b/i} + \boldsymbol{\omega}_{b/i} \times \left[\mathbf{I}_b \boldsymbol{\omega}_{b/i} + \mathbf{I}_w \boldsymbol{\omega}_{w/b} \right]. \tag{3.3.12}$$

Thus, the angular acceleration of the booster is

$$\dot{\boldsymbol{\omega}}_{b/i} = \mathbf{I}_b^{-1} \left[\mathbf{m}_b - \dot{\mathbf{I}} \boldsymbol{\omega}_{b/i} - \boldsymbol{\omega}_{b/i} \times \left(\mathbf{I}_b \boldsymbol{\omega}_{b/i} + \mathbf{I}_w \boldsymbol{\omega}_{w/b} \right) \right]. \tag{3.3.13}$$

The rotation of the wing adds two additional terms to the angular acceleration that differ from the typical derivation for angular acceleration, such as in [8], which are

$$\mathbf{m}_w = -\dot{\mathbf{I}}_b \boldsymbol{\omega}_{b/i} - \boldsymbol{\omega}_{b/i} \times \mathbf{I}_w \boldsymbol{\omega}_{w/b}. \tag{3.3.14}$$

The first term in 3.3.14 results from the rate of change in the vehicles inertia as the wing deploys, and the second term is the gyroscopic effect for a rigid body that has a rotating subsystem. This moment due to the deployment of the wing will be considered as a disturbance in the control system design by making the assumption that it is small. The term $\dot{\mathbf{I}}_b$ is found by differentiating Equation 3.3.6, giving

$$\dot{\mathbf{I}}_b = \dot{\mathbf{I}}_w + \dot{\mathbf{I}}_f = \dot{\mathbf{I}}_{w,0}, \tag{3.3.15}$$

since the inertia of the fuselage remains constant and the location of the wing centre of gravity relative to the booster centre of gravity is assumed constant. To simplify the analysis, the wing is modelled as a box, with length equal to the wing span at zero wing sweep, b_0, width equal to the mean aerodynamic chord and zero wing sweep, c_0, and height equal to the maximum thickness, t. This assumption overestimates the inertia of the wing since the maximum thickness is used for the height, and so the modelled moment induced by the wing rotation will be larger than the true case to represent the worst-case scenario. The moments and cross products of wing inertia are

$$\begin{aligned}
I_{xx,w} &= \frac{m}{12}\left(c_0^2\sin(\Lambda)^2 + b_0^2\cos(\Lambda)^2 + t^2\right),\\
I_{yy,w} &= \frac{m}{12}\left(c_0^2\cos(\Lambda)^2 + b_0^2\sin(\Lambda)^2 + t^2\right),\\
I_{zz,w} &= \frac{m}{12}\left(c_0^2 + b_0^2\right),\\
I_{xy,w} &= \frac{m}{24}\left(b_0^2 - c_0^2\right)\sin(\Lambda),\\
I_{xz,w} &= 0,\\
I_{yz,w} &= 0.
\end{aligned} \tag{3.3.16}$$

The rate of change of the moments and cross products of inertia of the wing are found by differentiating the moments and cross products of inertia with respect to the wing sweep angle, resulting in

$$\begin{aligned}
\dot{I}_{xx,w} &= \frac{m\dot{\Lambda}}{12}\left(c_0^2 - b_0^2\right)\sin(2\Lambda),\\
\dot{I}_{yy,w} &= \frac{m\dot{\Lambda}}{12}\left(b_0^2 - c_0^2\right)\sin(2\Lambda),\\
\dot{I}_{zz,w} &= 0,\\
\dot{I}_{xy,w} &= \frac{m\dot{\Lambda}}{12}\left(b_0^2 - c_0^2\right)\cos(2\Lambda),\\
\dot{I}_{xz,w} &= 0,\\
\dot{I}_{yz,w} &= 0.
\end{aligned} \tag{3.3.17}$$

The inertia for the fuselage and the wing at zero sweep are provided in Appendix A, along with the mass and geometrical properties used to derive the inertias. The wing is deployed at a constant rate of $\dot{\Lambda} = 5^\circ s^{-1}$.

3.4 External Forces and Moments

The forces and moments in the kinetic equations result from the aerodynamic forces, $\mathbf{f}_a$, aerodynamic moments, $\mathbf{m}_a$, and the gravitational force, $\mathbf{f}_g$. The external forces are expanded as

$$\mathbf{f}_b = \begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix} = \mathbf{f}_a + \mathbf{f}_g, \tag{3.4.1}$$

and the external moments are expanded as

$$\mathbf{m}_b = \begin{bmatrix} L_b \\ M_b \\ N_b \end{bmatrix} = \mathbf{m}_a. \tag{3.4.2}$$

3.4.1 Aerodynamic

The aerodynamic forces and moments are derived from Bernoulli's principle for flow around an object, as well as incompressible flow (Mach less than 0.3). The forces and moments are proportional to the free-stream dynamic pressure, defined as

$$Q_\infty(h) = \frac{1}{2}\rho_\infty(h)V_\infty^2, \tag{3.4.3}$$

where $\rho_\infty(h)$ is the air density that is a function of the altitude above mean sea level, h. For the troposphere ($h \leq 11km$), $\rho_\infty(h)$ is

$$\rho_\infty(h) = \rho_0\left(1 - \frac{l}{T_0}h\right)^{\frac{g_0}{lR_a}-1} \tag{3.4.4}$$

where $T_0 = 288.15K$ is the air temperature at sea level, $g_0 = 9.8067m/s^2$ is the gravitational acceleration at sea level, $\rho_0 = 1.225kg/m^3$ is the air density at sea level, $l = 6.5°C/m$ is the troposphere lapse rate and $R_a = 286.9J/kg/K$ is the ideal gas constant for air [35]. The aerodynamic forces are

$$\mathbf{f}_\mathcal{B}(\Lambda, h, M, \alpha, \beta, \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}, \boldsymbol{\delta}) = Q_\infty(h)S \begin{bmatrix} C_X(\Lambda, M, \alpha, \beta, \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}, \boldsymbol{\delta}) \\ C_Y(\Lambda, M, \alpha, \beta, \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}, \boldsymbol{\delta}) \\ C_Z(\Lambda, M, \alpha, \beta, \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}, \boldsymbol{\delta}) \end{bmatrix}, \tag{3.4.5}$$

where S is the wing planform area and C_X, C_Y and C_Z are non-dimensional aerodynamic force coefficients that encapsulate the aerodynamics of a specific airframe, and are therefore a function of the wing sweep angle. Furthermore, these coefficients vary with the compressibility of air, and are therefore also functions

of the Mach number. Similarly, the aerodynamic moments are given by

$$\mathbf{m}_{\mathcal{B}}(\Lambda,h,M,\alpha,\beta,\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}},\boldsymbol{\delta}) = Q_{\infty}(h)S \begin{bmatrix} b(\Lambda) & 0 & 0 \\ 0 & c(\Lambda) & 0 \\ 0 & 0 & b(\Lambda) \end{bmatrix} \begin{bmatrix} C_l(\Lambda,M,\alpha,\beta,\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}},\boldsymbol{\delta}) \\ C_m(\Lambda,M,\alpha,\beta,\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}},\boldsymbol{\delta}) \\ C_n(\Lambda,M,\alpha,\beta,\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}},\boldsymbol{\delta}) \end{bmatrix}, \tag{3.4.6}$$

where C_l, C_m and C_n are non-dimensional moment coefficients. The wing span varies with the wing sweep angle as

$$b(\Lambda) = \begin{cases} b_0 \cos(\Lambda) & \text{if} \quad 0° \leq \Lambda \leq 60°, \\ b_t & \text{if} \quad 60° < \Lambda \leq 90°, \end{cases} \tag{3.4.7}$$

where b_t is the tail span. Similarly, the wing chord varies with the wing sweep angle as

$$c(\Lambda) = \begin{cases} c_0 \sec(\Lambda) & \text{if} \quad 0° \leq \Lambda \leq 60°, \\ c_t & \text{if} \quad 60° < \Lambda \leq 90°, \end{cases} \tag{3.4.8}$$

where c_t is the tail chord. The aerodynamic coefficients are required in the body frame, however, the aerodynamic analysis software models the airflow in the stability frame to provide the lift coefficient, C_L, drag coefficient, C_D, and stability roll and yaw moment coefficients $C_{l,s}$ and $C_{n,s}$, respectively. The coefficients in the stability frame are rotated into the body frame using the DCM from Equation 3.1.6, giving

$$\begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} = \left[\mathbf{A}_{\mathcal{B}}^{\mathcal{S}}\right]^T \begin{bmatrix} -C_D \\ C_Y \\ -C_L \end{bmatrix} = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} -C_D \\ C_Y \\ -C_L \end{bmatrix} \tag{3.4.9}$$

and

$$\begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} = \left[\mathbf{A}_{\mathcal{B}}^{\mathcal{S}}\right]^T \begin{bmatrix} C_{l,s} \\ C_m \\ C_{n,s} \end{bmatrix} = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} C_{l,s} \\ C_m \\ C_{n,s} \end{bmatrix}. \tag{3.4.10}$$

For small flow incidence angles, the aerodynamic force and moment coefficients may be expanded as [36]

$$\begin{bmatrix} C_Y \\ C_L \end{bmatrix} = \begin{bmatrix} C_{Y_0} \\ C_{L_0} \end{bmatrix} + \begin{bmatrix} C_{Y_\alpha} & C_{Y_\beta} \\ C_{L_\alpha} & C_{L_\beta} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} + \frac{1}{2V_\infty} \begin{bmatrix} C_{Y_P} & C_{Y_Q} & C_{Y_R} \\ C_{L_P} & C_{L_Q} & C_{L_R} \end{bmatrix} \begin{bmatrix} b & 0 & 0 \\ 0 & c & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \begin{bmatrix} C_{Y_{\delta_{A,R}}} & C_{Y_{\delta_{A,L}}} \\ C_{L_{\delta_{A,R}}} & C_{L_{\delta_{A,L}}} \end{bmatrix} \begin{bmatrix} \delta_{A,R} \\ \delta_{A,L} \end{bmatrix} + \begin{bmatrix} C_{Y_{\delta_{E,R}}} & C_{Y_{\delta_{E,L}}} \\ C_{L_{\delta_{E,R}}} & C_{L_{\delta_{E,L}}} \end{bmatrix} \begin{bmatrix} \delta_{E,R} \\ \delta_{E,L} \end{bmatrix} + \begin{bmatrix} C_{Y_{\delta_{R,R}}} & C_{Y_{\delta_{R,L}}} \\ C_{L_{\delta_{R,R}}} & C_{L_{\delta_{R,L}}} \end{bmatrix} \begin{bmatrix} \delta_{R,R} \\ \delta_{R,L} \end{bmatrix} \tag{3.4.11}$$

and

$$\begin{bmatrix} C_{l,s} \\ C_m \\ C_{n,s} \end{bmatrix} = \begin{bmatrix} C_{l_0} \\ C_{m_0} \\ C_{n_0} \end{bmatrix} + \begin{bmatrix} C_{l_\alpha} & C_{l_\beta} \\ C_{m_\alpha} & C_{m_\beta} \\ C_{n_\alpha} & C_{n_\beta} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} + \frac{1}{2V_\infty} \begin{bmatrix} C_{l_P} & C_{l_Q} & C_{l_R} \\ C_{m_P} & C_{m_Q} & C_{m_R} \\ C_{n_P} & C_{n_Q} & C_{n_R} \end{bmatrix} \begin{bmatrix} b & 0 & 0 \\ 0 & c & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \begin{bmatrix} C_{l_{\delta_{A,R}}} & C_{l_{\delta_{A,L}}} \\ C_{m_{\delta_{A,R}}} & C_{m_{\delta_{A,L}}} \\ C_{n_{\delta_{A,R}}} & C_{n_{\delta_{A,L}}} \end{bmatrix} \begin{bmatrix} \delta_{A,R} \\ \delta_{A,L} \end{bmatrix} + \begin{bmatrix} C_{l_{\delta_{E,R}}} & C_{l_{\delta_{E,L}}} \\ C_{m_{\delta_{E,R}}} & C_{m_{\delta_{E,L}}} \\ C_{n_{\delta_{E,R}}} & C_{n_{\delta_{E,L}}} \end{bmatrix} \begin{bmatrix} \delta_{E,R} \\ \delta_{E,L} \end{bmatrix} + \begin{bmatrix} C_{l_{\delta_{R,R}}} & C_{l_{\delta_{R,L}}} \\ C_{m_{\delta_{R,R}}} & C_{m_{\delta_{R,L}}} \\ C_{n_{\delta_{R,R}}} & C_{n_{\delta_{R,L}}} \end{bmatrix} \begin{bmatrix} \delta_{R,R} \\ \delta_{R,L} \end{bmatrix}, \tag{3.4.12}$$

where C_{L_0}, C_{Y_0}, C_{l_0}, C_{m_0} and C_{n_0} are the static force and moment coefficients. The non-dimensional stability and control derivatives are the terms of the form C_{A_B}, defined as

$$C_{A_B} \equiv \frac{\partial C_A}{\partial B}, \tag{3.4.13}$$

for the sets $A \in \{L, Y, l, m, n\}$ and $B \in \{\alpha, \beta, P, Q, R, \delta_{A,R}, \delta_{A,L}, \delta_{E,R}, \delta_{E,L}, \delta_{R,R}, \delta_{R,L}\}$. In order to obtain these stability and control derivatives, the potential flow panel method software PANUKL is used to obtain the aerodynamic lift, side force and moment coefficients for perturbations in the flow conditions. For example, C_{L_α} is computed as

$$C_{L_\alpha} = \frac{C_L(\alpha = 2.5°) - C_L(\alpha = 0)}{2.5(\pi/180)}, \tag{3.4.14}$$

where the factor $\pi/180$ converts degrees to radians in order to preserve the non-dimensionality of the stability and control derivatives. Potential flow solvers such as PANUKL are only valid for incompressible flow below Mach 0.3. The Prandtl-Glauert compressibility correction is therefore applied to the resulting stability and control derivatives for a range of subsonic airspeeds up to Mach 0.8, at which point compressibility corrections are no longer valid to a drastic change in the underlying physics that occurs for

transonic airflows ($0.8 < M < 1.2$) [37]. The Prandtl-Glauert compressibility correction is

$$C_P = \frac{C_{P,0}}{\sqrt{1-M^2}}, \tag{3.4.15}$$

where $C_{P,0}$ is the pressure coefficient for $M < 0.3$. This correction is applicable to force and moment coefficients as well, and is therefore used to obtain the stability derivatives as a function of the Mach number up to transonic flow. The drag coefficient is modelled as

$$C_D = C_{D_0} + \frac{C_L^2}{\pi \mathcal{R}(\Lambda) e(\Lambda)}, \tag{3.4.16}$$

where C_{D_0} is the parasitic drag coefficient, $\mathcal{R}$ is the aspect ratio of the wing, and e is the Oswald efficiency factor of the wing. The aspect ratio is a function of the wing sweep angle as

$$\mathcal{R}(\Lambda) = \frac{b(\Lambda)}{c(\Lambda)}. \tag{3.4.17}$$

The Oswald efficiency factor varies with the wing sweep angle as [38]

$$e(\Lambda) = e_0 \cos(\Lambda) \tag{3.4.18}$$

where e_0 is the Oswald efficiency factor of the wing at zero wing sweep. For square wings $e_0 = 0.7$ [39], which is used in this work. For wing sweep angles between 60° and 90°, a efficiency factor for the wing at 60° is used ($e_0 = 0.35$). The first term in Equation 3.4.16 is the parasitic drag, and the second term is the induced drag that results from downwash on a finite wing tilting the lift vector in the negative $\bar{x}_S$ direction.

Aerodynamic Database

Using Equations 3.4.13-3.4.16, an aerodynamic database is constructed for $\Lambda \in \{0, 15, 30, 45, 60, 90\}$ and $M \leq 0.8$. Figures 3.7-3.11 shows the static aerodynamic coefficients as a function of Λ and α at Mach 0.4. As expected, the booster produces less lift at the higher sweep angles. The static roll moment produced in the oblique wing configuration is generally positive, resulting in a roll moment towards the forward swept wing, and increases with the angle of attack as expected. However, C_{l_0} is greatest at a sweep angle of 30° and is smallest, for the oblique wing configuration, at 60° sweep. Furthermore, significant side forces and yaw moments are produced at non-zero angles of attack in the oblique wing configuration.

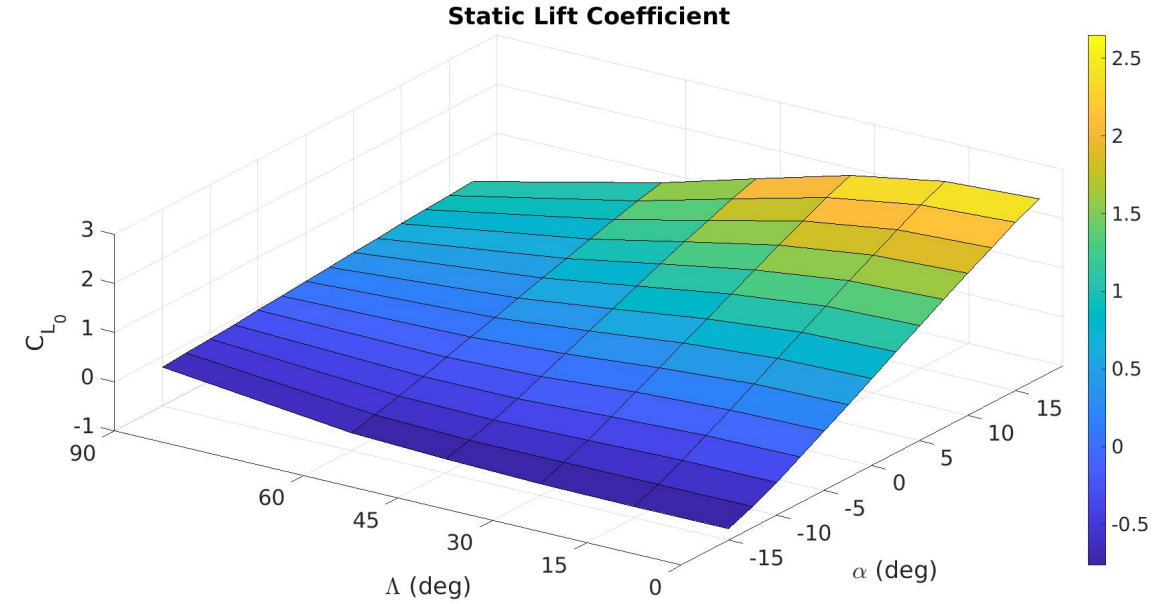


Figure 3.7: Static lift coefficient as a function of wing sweep angle and angle of attack.

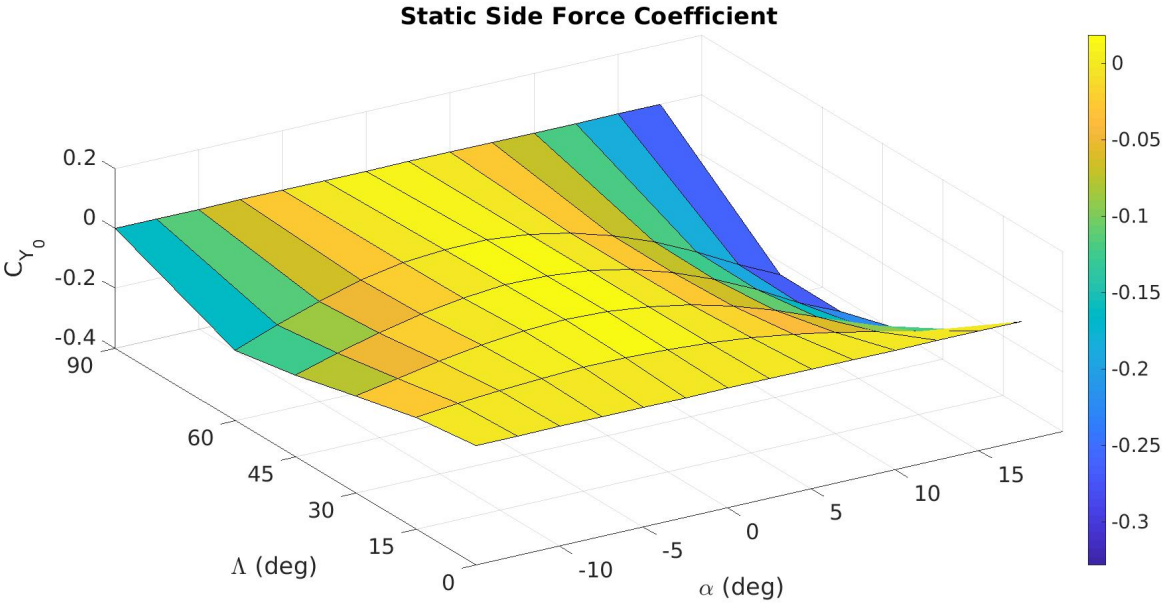


Figure 3.8: Static side force coefficient as a function of wing sweep angle and angle of attack.

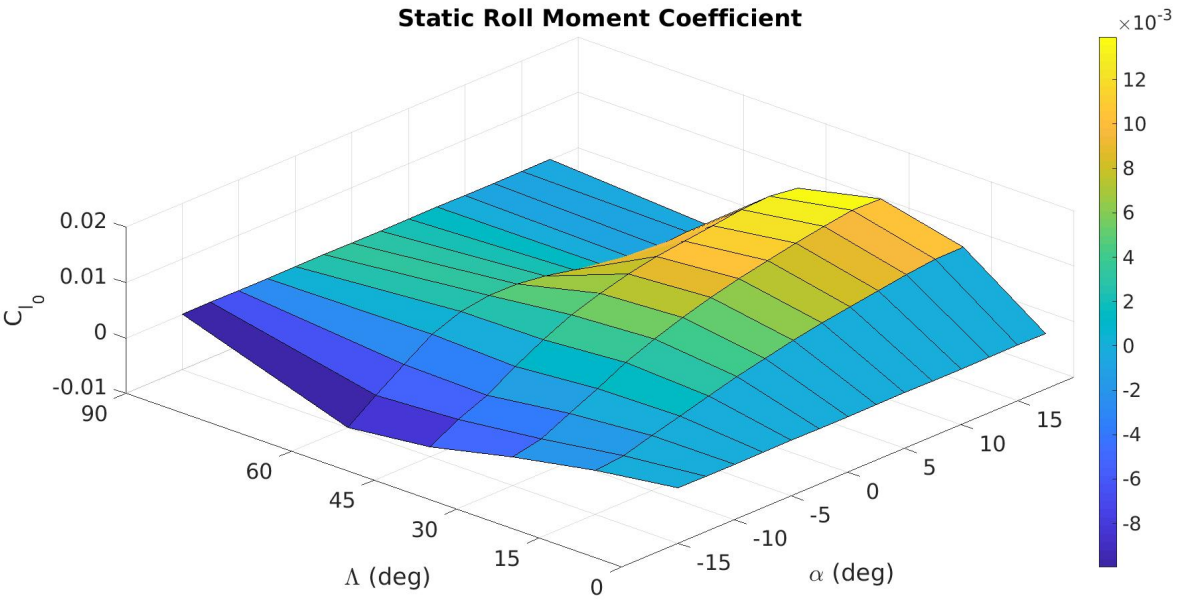


Figure 3.9: Static roll moment coefficient as a function of wing sweep angle and angle of attack.

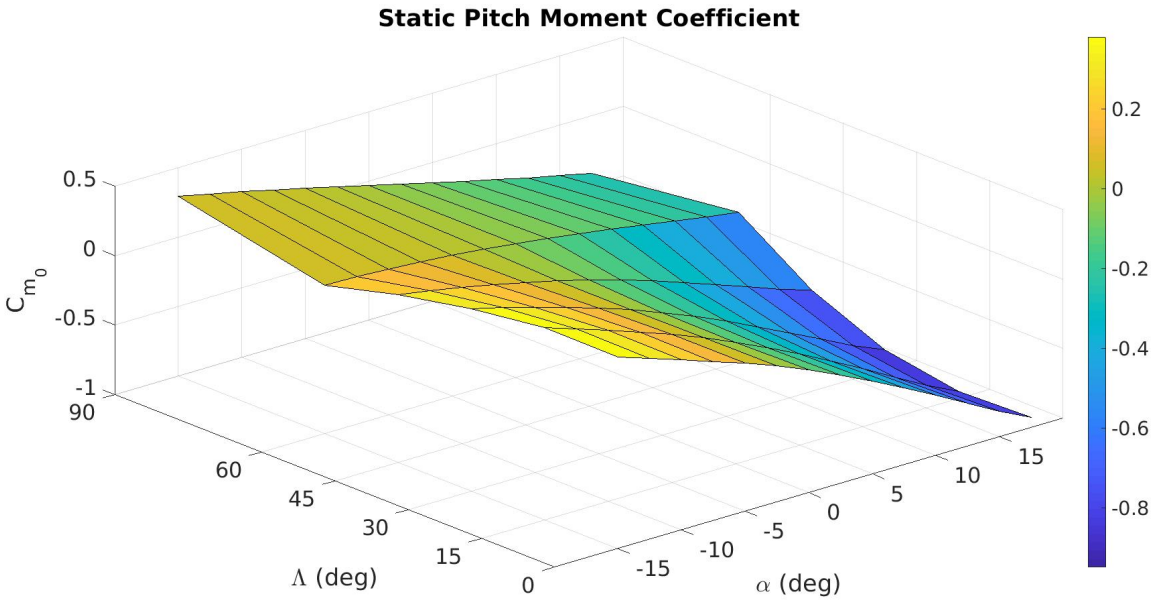


Figure 3.10: Static pitch moment coefficient as a function of wing sweep angle and angle of attack.

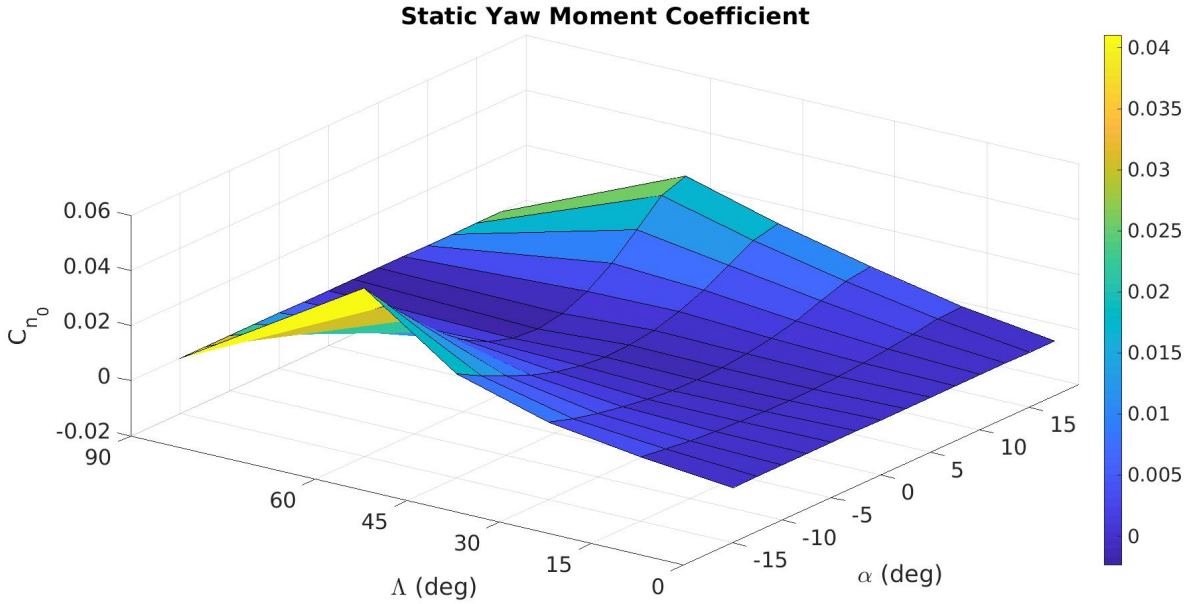


Figure 3.11: Static yaw moment coefficient as a function of wing sweep angle and angle of attack.

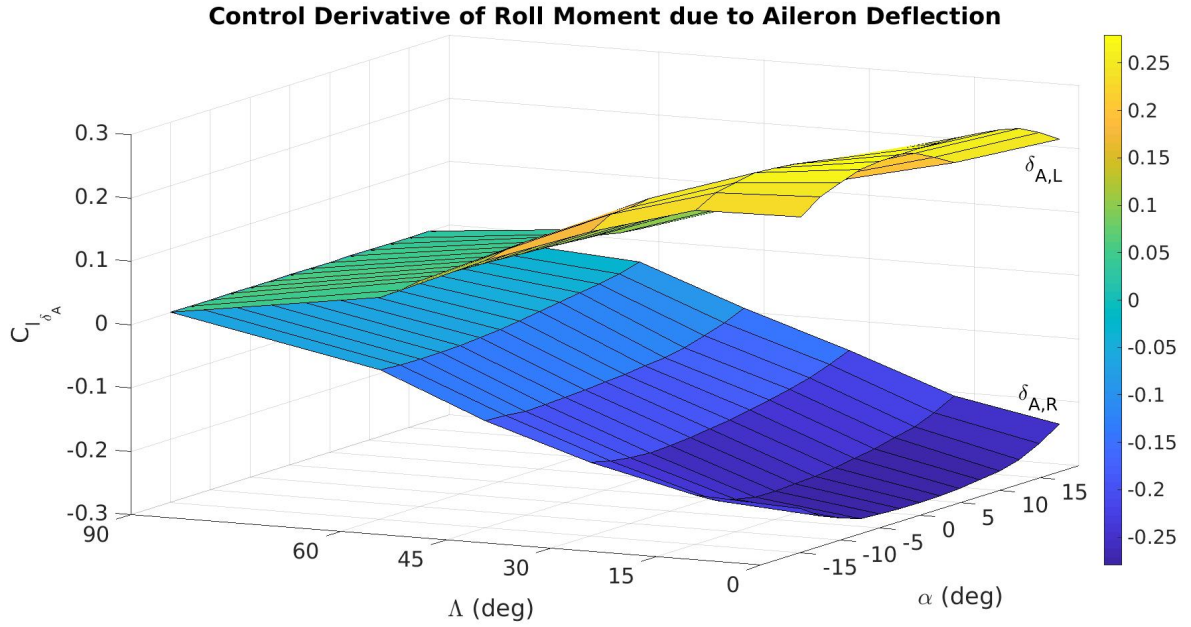


Figure 3.12: Control derivative of the roll moment due to aileron deflection as a function of wing sweep angle and angle of attack.

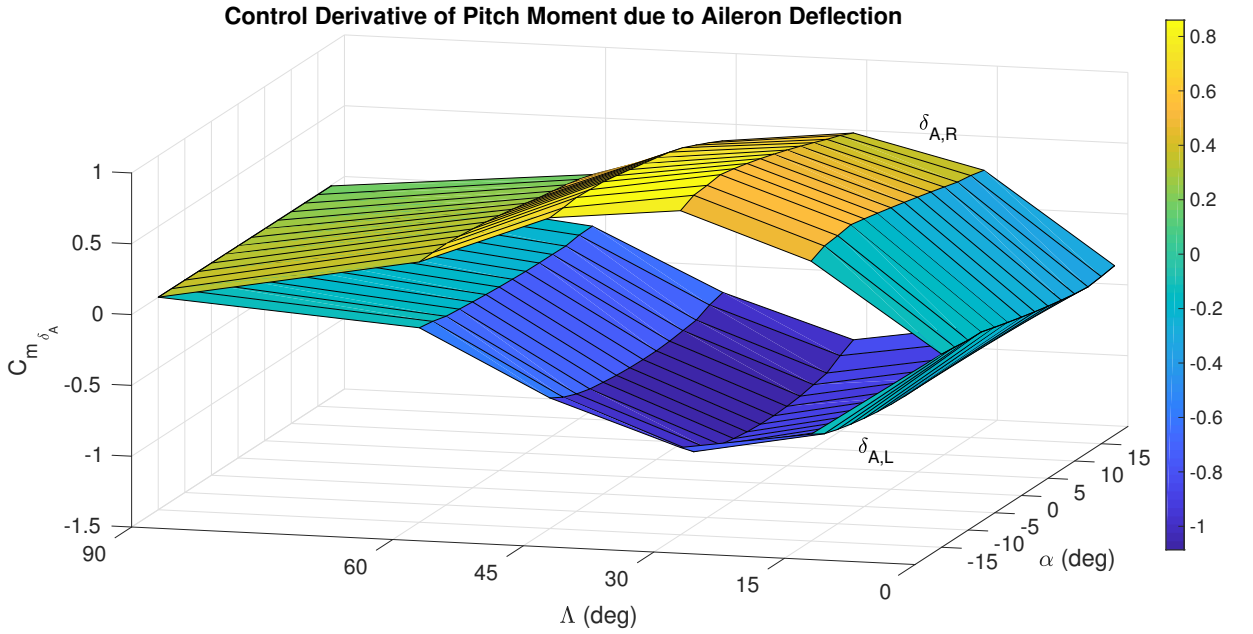


Figure 3.13: Control derivative of the pitch moment due to aileron deflection as a function of wing sweep angle and angle of attack.

Figure 3.12 shows $C_{l_{\delta_A}}$ as a function of the wing sweep angle and angle of attack, and exemplifies the reduced aileron control effectiveness resulting from sweeping the wing. $C_{l_{\delta_A}}$ is exactly zero at $\Lambda = 90°$ since the wing is stowed in the rocket configuration. Figure 3.13 shows $C_{m_{\delta_A}}$ as a function of the wing sweep angle and angle of attack, showing the cross-coupling between lateral and longitudinal motions due to aileron deflections.

3.4.2 Gravitational

The gravitational force is positive in the inertial down direction, and is expressed in body coordinates by multiplying the inertial weight vector by the DCM as

$$\begin{bmatrix} X_G \\ Y_G \\ Z_G \end{bmatrix} = \mathbf{A}_{\mathcal{I}}^{\mathcal{B}} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = mg \begin{bmatrix} -\sin(\Theta) \\ \cos(\Theta)\sin(\Phi) \\ \cos(\Theta)\cos(\Phi) \end{bmatrix}. \tag{3.4.19}$$

A constant gravity field is assumed, with gravitational acceleration $g = 9.81\ m/s^2$ acting in the positive down direction of the inertial coordinate system.

3.5 Equations of Motion Simplification

The nonlinear equations of motion for a rigid body have been derived in the previous sections of this chapter. This section reduces these equations to a simplified linear set to reduce the complexity of the controller design. The equations of motion are restated and expanded here. Treating the wind velocity as

a disturbance, the position dynamics from Equation 3.2.6 are given as

$$\begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \begin{bmatrix} \cos\Theta\cos\Psi & \sin\Phi\sin\Theta\cos\Psi - \cos\Phi\sin\Phi & \cos\Phi\sin\Theta\cos\Psi + \cos\Phi\cos\Phi \\ \cos\Theta\sin\Psi & \sin\Phi\sin\Theta\sin\Psi + \cos\Phi\cos\Phi & \cos\Phi\sin\Theta\sin\Psi - \sin\Phi\cos\Phi \\ -\sin\Theta & \sin\Phi\cos\Theta & \cos\Phi\cos\Theta \end{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix}. \tag{3.5.1}$$

The attitude dynamics for Euler angles are restated from Equation 3.2.13 as

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \sin\Phi\sec\Theta & \cos\Phi\sec\Theta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix}. \tag{3.5.2}$$

Since the wind velocity is treated as a disturbance, the external forces will directly affect $\mathbf{v}_{\mathcal{B}/\mathcal{A}}$ and the dynamics from Equation 3.3.3 are

$$\begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \end{bmatrix} = \frac{Q_\infty S}{m} \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} -C_D \\ C_Y \\ -C_L \end{bmatrix} + g \begin{bmatrix} -\sin(\Theta) \\ \cos(\Theta)\sin(\Phi) \\ \cos(\Theta)\cos(\Phi) \end{bmatrix} - \begin{bmatrix} 0 & -R & Q \\ R & 0 & -P \\ -Q & P & 0 \end{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix}. \tag{3.5.3}$$

The moments due to the wing rotation given in Equation 3.3.14 are treated as a disturbance in the angular velocity dynamics, thus Equation 3.3.13 reduces to

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}^{-1} \left(Q_\infty S \begin{bmatrix} b & 0 & 0 \\ 0 & c & 0 \\ 0 & 0 & b \end{bmatrix} \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} C_{l,s} \\ C_m \\ C_{n,s} \end{bmatrix} - \begin{bmatrix} 0 & -R & Q \\ R & 0 & -P \\ -Q & P & 0 \end{bmatrix} \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \right). \tag{3.5.4}$$

3.5.1 Linearisation

Twelve nonlinear, coupled equations are required to model the equations of motion of the booster in the inertial frame. To simplify the description of the booster motion, we first note that the yaw angle only couples into the position dynamics, and the position of the booster itself only loosely couples back into the other dynamics due to the variation of air density with altitude. By assuming that the change in air

density is slow with respect to the rest of the dynamics, the state of the booster may be chosen as

$$\mathbf{x} = \begin{bmatrix} U & V & W & P & Q & R & \Phi & \Theta \end{bmatrix}^T. \tag{3.5.5}$$

The control input is the deflection of the six independently actuated control surfaces,

$$\mathbf{u} = \begin{bmatrix} \delta_{A,R} & \delta_{A,L} & \delta_{E,R} & \delta_{E,L} & \delta_{R,R} & \delta_{R,L} \end{bmatrix}^T. \tag{3.5.6}$$

The nonlinear dynamics are thus described in state space form as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}). \tag{3.5.7}$$

Considering the state and control as the sum of a trim value and the perturbation about the trim value yields

$$\mathbf{x} = \mathbf{x}_T + \Delta\mathbf{x} \tag{3.5.8}$$

and

$$\mathbf{u} = \mathbf{u}_T + \Delta\mathbf{u}, \tag{3.5.9}$$

where the subscript T denotes the trim condition. The perturbation states and inputs are denoted as

$$\Delta\mathbf{x} = \begin{bmatrix} u & v & w & p & q & r & \phi & \theta \end{bmatrix}^T \tag{3.5.10}$$

and

$$\Delta\mathbf{u} = \begin{bmatrix} \delta_{a,r} & \delta_{a,l} & \delta_{e,r} & \delta_{e,l} & \delta_{r,r} & \delta_{r,l} \end{bmatrix}^T, \tag{3.5.11}$$

respectively. Taking the Taylor series approximation of Equation 3.5.7 yields

$$\dot{\mathbf{x}}_T + \Delta\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}_T + \Delta\mathbf{x}, \mathbf{u}_T + \Delta\mathbf{u}) = \mathbf{f}(\mathbf{x}_T, \mathbf{u}_T) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_T \Delta\mathbf{x} + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_T \Delta\mathbf{u} + H.O.T. \tag{3.5.12}$$

The higher order terms (H.O.T.) of the Taylor series are dropped by assuming the deviations from trim are small, so the dynamics about trim are approximated by

$$\Delta\dot{\mathbf{x}} \approx \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_T \Delta\mathbf{x} + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_T \Delta\mathbf{u} = \mathbf{A}\Delta\mathbf{x} + \mathbf{B}\Delta\mathbf{u}. \tag{3.5.13}$$

Equation 3.5.13 is expanded as

$$\Delta\dot{\mathbf{x}} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{\partial \dot{U}}{\partial U} & \frac{\partial \dot{U}}{\partial V} & \frac{\partial \dot{U}}{\partial W} & \frac{\partial \dot{U}}{\partial P} & \frac{\partial \dot{U}}{\partial Q} & \frac{\partial \dot{U}}{\partial R} & \frac{\partial \dot{U}}{\partial \Phi} & \frac{\partial \dot{U}}{\partial \Theta} \\ \frac{\partial \dot{V}}{\partial U} & \frac{\partial \dot{V}}{\partial V} & \frac{\partial \dot{V}}{\partial W} & \frac{\partial \dot{V}}{\partial P} & \frac{\partial \dot{V}}{\partial Q} & \frac{\partial \dot{V}}{\partial R} & \frac{\partial \dot{V}}{\partial \Phi} & \frac{\partial \dot{V}}{\partial \Theta} \\ \frac{\partial \dot{W}}{\partial U} & \frac{\partial \dot{W}}{\partial V} & \frac{\partial \dot{W}}{\partial W} & \frac{\partial \dot{W}}{\partial P} & \frac{\partial \dot{W}}{\partial Q} & \frac{\partial \dot{W}}{\partial R} & \frac{\partial \dot{W}}{\partial \Phi} & \frac{\partial \dot{W}}{\partial \Theta} \\ \frac{\partial \dot{P}}{\partial U} & \frac{\partial \dot{P}}{\partial V} & \frac{\partial \dot{P}}{\partial W} & \frac{\partial \dot{P}}{\partial P} & \frac{\partial \dot{P}}{\partial Q} & \frac{\partial \dot{P}}{\partial R} & \frac{\partial \dot{P}}{\partial \Phi} & \frac{\partial \dot{P}}{\partial \Theta} \\ \frac{\partial \dot{Q}}{\partial U} & \frac{\partial \dot{Q}}{\partial V} & \frac{\partial \dot{Q}}{\partial W} & \frac{\partial \dot{Q}}{\partial P} & \frac{\partial \dot{Q}}{\partial Q} & \frac{\partial \dot{Q}}{\partial R} & \frac{\partial \dot{Q}}{\partial \Phi} & \frac{\partial \dot{Q}}{\partial \Theta} \\ \frac{\partial \dot{R}}{\partial U} & \frac{\partial \dot{R}}{\partial V} & \frac{\partial \dot{R}}{\partial W} & \frac{\partial \dot{R}}{\partial P} & \frac{\partial \dot{R}}{\partial Q} & \frac{\partial \dot{R}}{\partial R} & \frac{\partial \dot{R}}{\partial \Phi} & \frac{\partial \dot{R}}{\partial \Theta} \\ \frac{\partial \dot{\Phi}}{\partial U} & \frac{\partial \dot{\Phi}}{\partial V} & \frac{\partial \dot{\Phi}}{\partial W} & \frac{\partial \dot{\Phi}}{\partial P} & \frac{\partial \dot{\Phi}}{\partial Q} & \frac{\partial \dot{\Phi}}{\partial R} & \frac{\partial \dot{\Phi}}{\partial \Phi} & \frac{\partial \dot{\Phi}}{\partial \Theta} \\ \frac{\partial \dot{\Theta}}{\partial U} & \frac{\partial \dot{\Theta}}{\partial V} & \frac{\partial \dot{\Theta}}{\partial W} & \frac{\partial \dot{\Theta}}{\partial P} & \frac{\partial \dot{\Theta}}{\partial Q} & \frac{\partial \dot{\Theta}}{\partial R} & \frac{\partial \dot{\Theta}}{\partial \Phi} & \frac{\partial \dot{\Theta}}{\partial \Theta} \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \\ \phi \\ \theta \end{bmatrix}$$
$$+ \begin{bmatrix} \frac{\partial \dot{U}}{\partial \delta_{A,R}} & \frac{\partial \dot{U}}{\partial \delta_{A,L}} & \frac{\partial \dot{U}}{\partial \delta_{E,R}} & \frac{\partial \dot{U}}{\partial \delta_{E,L}} & \frac{\partial \dot{U}}{\partial \delta_{R,R}} & \frac{\partial \dot{U}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{V}}{\partial \delta_{A,R}} & \frac{\partial \dot{V}}{\partial \delta_{A,L}} & \frac{\partial \dot{V}}{\partial \delta_{E,R}} & \frac{\partial \dot{V}}{\partial \delta_{E,L}} & \frac{\partial \dot{V}}{\partial \delta_{R,R}} & \frac{\partial \dot{V}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{W}}{\partial \delta_{A,R}} & \frac{\partial \dot{W}}{\partial \delta_{A,L}} & \frac{\partial \dot{W}}{\partial \delta_{E,R}} & \frac{\partial \dot{W}}{\partial \delta_{E,L}} & \frac{\partial \dot{W}}{\partial \delta_{R,R}} & \frac{\partial \dot{W}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{P}}{\partial \delta_{A,R}} & \frac{\partial \dot{P}}{\partial \delta_{A,L}} & \frac{\partial \dot{P}}{\partial \delta_{E,R}} & \frac{\partial \dot{P}}{\partial \delta_{E,L}} & \frac{\partial \dot{P}}{\partial \delta_{R,R}} & \frac{\partial \dot{P}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{Q}}{\partial \delta_{A,R}} & \frac{\partial \dot{Q}}{\partial \delta_{A,L}} & \frac{\partial \dot{Q}}{\partial \delta_{E,R}} & \frac{\partial \dot{Q}}{\partial \delta_{E,L}} & \frac{\partial \dot{Q}}{\partial \delta_{R,R}} & \frac{\partial \dot{Q}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{R}}{\partial \delta_{A,R}} & \frac{\partial \dot{R}}{\partial \delta_{A,L}} & \frac{\partial \dot{R}}{\partial \delta_{E,R}} & \frac{\partial \dot{R}}{\partial \delta_{E,L}} & \frac{\partial \dot{R}}{\partial \delta_{R,R}} & \frac{\partial \dot{R}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{\Phi}}{\partial \delta_{A,R}} & \frac{\partial \dot{\Phi}}{\partial \delta_{A,L}} & \frac{\partial \dot{\Phi}}{\partial \delta_{E,R}} & \frac{\partial \dot{\Phi}}{\partial \delta_{E,L}} & \frac{\partial \dot{\Phi}}{\partial \delta_{R,R}} & \frac{\partial \dot{\Phi}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{\Theta}}{\partial \delta_{A,R}} & \frac{\partial \dot{\Theta}}{\partial \delta_{A,L}} & \frac{\partial \dot{\Theta}}{\partial \delta_{E,R}} & \frac{\partial \dot{\Theta}}{\partial \delta_{E,L}} & \frac{\partial \dot{\Theta}}{\partial \delta_{R,R}} & \frac{\partial \dot{\Theta}}{\partial \delta_{R,L}} \end{bmatrix} \begin{bmatrix} \delta_{a,r} \\ \delta_{a,l} \\ \delta_{e,r} \\ \delta_{e,l} \\ \delta_{r,r} \\ \delta_{r,l} \end{bmatrix}. \quad (3.5.14)$$

It is desirable to work with α and β instead of V and W, and then to normalise the perturbation velocity along the body $\bar{x}_{\mathcal{B}}$ axis by the airspeed. For small incidence angles, the component of velocity along the $\bar{x}_{\mathcal{B}}$ axis is dominant, thus Equation 3.2.7 simplifies to

$$V_\infty = \sqrt{U^2 + V^2 + W^2} \approx U = \frac{U}{V_\infty} V_\infty. \quad (3.5.15)$$

Furthermore, Equation 3.2.8 may be approximated by

$$\tan\alpha = \frac{W}{U} \approx \alpha \Rightarrow W = \alpha V_\infty, \quad (3.5.16)$$

and Equation 3.2.9 may be approximated by

$$\sin\beta = \frac{V}{V_\infty} \approx \beta \Rightarrow V = \beta V_\infty \quad (3.5.17)$$

Substituting these relations into Equation 3.5.14 and dividing through by V_∞ gives

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{u}/V_\infty \\ \dot{\beta} \\ \dot{\alpha} \\ \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{\partial \dot{U}}{\partial U} & \frac{\partial \dot{U}}{\partial V} & \frac{\partial \dot{U}}{\partial W} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial P} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial Q} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial R} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \Phi} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \Theta} \\ \frac{\partial \dot{V}}{\partial U} & \frac{\partial \dot{V}}{\partial V} & \frac{\partial \dot{V}}{\partial W} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial P} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial Q} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial R} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \Phi} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \Theta} \\ \frac{\partial \dot{W}}{\partial U} & \frac{\partial \dot{W}}{\partial V} & \frac{\partial \dot{W}}{\partial W} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial P} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial Q} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial R} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \Phi} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \Theta} \\ V_\infty\frac{\partial \dot{P}}{\partial U} & V_\infty\frac{\partial \dot{P}}{\partial V} & V_\infty\frac{\partial \dot{P}}{\partial W} & \frac{\partial \dot{P}}{\partial P} & \frac{\partial \dot{P}}{\partial Q} & \frac{\partial \dot{P}}{\partial R} & \frac{\partial \dot{P}}{\partial \Phi} & \frac{\partial \dot{P}}{\partial \Theta} \\ V_\infty\frac{\partial \dot{Q}}{\partial U} & V_\infty\frac{\partial \dot{Q}}{\partial V} & V_\infty\frac{\partial \dot{Q}}{\partial W} & \frac{\partial \dot{Q}}{\partial P} & \frac{\partial \dot{Q}}{\partial Q} & \frac{\partial \dot{Q}}{\partial R} & \frac{\partial \dot{Q}}{\partial \Phi} & \frac{\partial \dot{Q}}{\partial \Theta} \\ V_\infty\frac{\partial \dot{R}}{\partial U} & V_\infty\frac{\partial \dot{R}}{\partial V} & V_\infty\frac{\partial \dot{R}}{\partial W} & \frac{\partial \dot{R}}{\partial P} & \frac{\partial \dot{R}}{\partial Q} & \frac{\partial \dot{R}}{\partial R} & \frac{\partial \dot{R}}{\partial \Phi} & \frac{\partial \dot{R}}{\partial \Theta} \\ V_\infty\frac{\partial \dot{\Phi}}{\partial U} & V_\infty\frac{\partial \dot{\Phi}}{\partial V} & V_\infty\frac{\partial \dot{\Phi}}{\partial W} & \frac{\partial \dot{\Phi}}{\partial P} & \frac{\partial \dot{\Phi}}{\partial Q} & \frac{\partial \dot{\Phi}}{\partial R} & \frac{\partial \dot{\Phi}}{\partial \Phi} & \frac{\partial \dot{\Phi}}{\partial \Theta} \\ V_\infty\frac{\partial \dot{\Theta}}{\partial U} & V_\infty\frac{\partial \dot{\Theta}}{\partial V} & V_\infty\frac{\partial \dot{\Theta}}{\partial W} & \frac{\partial \dot{\Theta}}{\partial P} & \frac{\partial \dot{\Theta}}{\partial Q} & \frac{\partial \dot{\Theta}}{\partial R} & \frac{\partial \dot{\Theta}}{\partial \Phi} & \frac{\partial \dot{\Theta}}{\partial \Theta} \end{bmatrix} \begin{bmatrix} u/V_\infty \\ \beta \\ \alpha \\ p \\ q \\ r \\ \phi \\ \theta \end{bmatrix}$$
$$+ \begin{bmatrix} \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \delta_{A,R}} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \delta_{A,L}} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \delta_{E,R}} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \delta_{E,L}} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \delta_{R,R}} & \frac{1}{V_\infty}\frac{\partial \dot{U}}{\partial \delta_{R,L}} \\ \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \delta_{A,R}} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \delta_{A,L}} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \delta_{E,R}} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \delta_{E,L}} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \delta_{R,R}} & \frac{1}{V_\infty}\frac{\partial \dot{V}}{\partial \delta_{R,L}} \\ \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \delta_{A,R}} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \delta_{A,L}} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \delta_{E,R}} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \delta_{E,L}} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \delta_{R,R}} & \frac{1}{V_\infty}\frac{\partial \dot{W}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{P}}{\partial \delta_{A,R}} & \frac{\partial \dot{P}}{\partial \delta_{A,L}} & \frac{\partial \dot{P}}{\partial \delta_{E,R}} & \frac{\partial \dot{P}}{\partial \delta_{E,L}} & \frac{\partial \dot{P}}{\partial \delta_{R,R}} & \frac{\partial \dot{P}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{Q}}{\partial \delta_{A,R}} & \frac{\partial \dot{Q}}{\partial \delta_{A,L}} & \frac{\partial \dot{Q}}{\partial \delta_{E,R}} & \frac{\partial \dot{Q}}{\partial \delta_{E,L}} & \frac{\partial \dot{Q}}{\partial \delta_{R,R}} & \frac{\partial \dot{Q}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{R}}{\partial \delta_{A,R}} & \frac{\partial \dot{R}}{\partial \delta_{A,L}} & \frac{\partial \dot{R}}{\partial \delta_{E,R}} & \frac{\partial \dot{R}}{\partial \delta_{E,L}} & \frac{\partial \dot{R}}{\partial \delta_{R,R}} & \frac{\partial \dot{R}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{\Phi}}{\partial \delta_{A,R}} & \frac{\partial \dot{\Phi}}{\partial \delta_{A,L}} & \frac{\partial \dot{\Phi}}{\partial \delta_{E,R}} & \frac{\partial \dot{\Phi}}{\partial \delta_{E,L}} & \frac{\partial \dot{\Phi}}{\partial \delta_{R,R}} & \frac{\partial \dot{\Phi}}{\partial \delta_{R,L}} \\ \frac{\partial \dot{\Theta}}{\partial \delta_{A,R}} & \frac{\partial \dot{\Theta}}{\partial \delta_{A,L}} & \frac{\partial \dot{\Theta}}{\partial \delta_{E,R}} & \frac{\partial \dot{\Theta}}{\partial \delta_{E,L}} & \frac{\partial \dot{\Theta}}{\partial \delta_{R,R}} & \frac{\partial \dot{\Theta}}{\partial \delta_{R,L}} \end{bmatrix} \begin{bmatrix} \delta_{a,r} \\ \delta_{a,l} \\ \delta_{e,r} \\ \delta_{e,l} \\ \delta_{r,r} \\ \delta_{r,l} \end{bmatrix}. \quad (3.5.18)$$

3.5.2 Equilibrium

In order to linearise the dynamics, we require a solution to the equations of motion given by Equation 3.5.7. A convenient solution is the equilibrium condition, or

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = 0. \quad (3.5.19)$$

Determining an equilibrium condition requires solving for 15 variables, the eight states, altitude and six control inputs, that results in zero forces and moments. Thus, this is an underdefined system without an unique solution. To reduce the number of unknown variables, we note that it is desirable for the booster to fly at zero sideslip. This reduces the drag experienced by the booster, as well as helps prevent the natural tendency of the oblique wing to unsweep itself as described in the literature. Furthermore, an equilibrium condition will be determined for a desired altitude and airspeed. This allows the air density and the stability and control derivatives to be considered as parameters. Finally, the six control inputs are grouped

into traditional aileron, δ_A, elevator, δ_E, and rudder, δ_R, control deflections as

$$\begin{bmatrix} \delta_A \\ \delta_E \\ \delta_R \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} \delta_{A,R} \\ \delta_{A,L} \\ \delta_{E,R} \\ \delta_{E,L} \\ \delta_{R,R} \\ \delta_{R,L} \end{bmatrix} \tag{3.5.20}$$

in the rocket configuration, and

$$\begin{bmatrix} \delta_A \\ \delta_E \\ \delta_R \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} \delta_{A,R} \\ \delta_{A,L} \\ \delta_{E,R} \\ \delta_{E,L} \\ \delta_{R,R} \\ \delta_{R,L} \end{bmatrix} \tag{3.5.21}$$

in the aircraft configuration. This results in six unknown states (α, Φ, Θ, δ_A, δ_E, δ_R) to be determined for an equilibrium condition. To determine the values of these that result in zero forces and moments, the multivariate Newton-Raphson iteration is used. Pseudo code for the implementation of the Newton-Raphson solver is provided in Algorithm 1, where

$$\mathbf{x} = \begin{bmatrix} \alpha & \Phi & \Theta & \delta_A & \delta_E & \delta_R \end{bmatrix}^T, \tag{3.5.22}$$

and

$$\mathbf{f} = \begin{bmatrix} X_b & Y_b & Z_b & L_b & M_b & N_b \end{bmatrix}^T. \tag{3.5.23}$$

Once a trim condition $\hat{\mathbf{x}}$ is determined, the individual control surface deflections are then computed by inverting Equation 3.5.20 and Equation 3.5.21 for the rocket configuration and aircraft configuration, respectively. Note that a check on the angle of attack is required to ensure that a solution is not determined for an angle of attack outside the linear range of the airfoil. The aerodynamic database constructed from PANUKL is used to obtain equilibrium conditions for each of the wing sweep angles considered, airspeeds up to Mach 0.8 and altitudes up to 11 km, which represents the entire range of the troposphere. Figure 3.14 illustrates the differences resulting from varying the wing sweep angle by displaying the trim angle of attack for wing sweep angles of 0°, 15°, 30° and 60° as a function of the altitude and Mach number. Each surface represents the trim angle of attack for the sweep angle as labelled. As the wing sweep increases, the wing produces less lift and therefore requires a larger angle of attack than lower sweep angles to maintain

trim at a given condition. This allows the booster to fly at equilibrium at a higher velocity and altitude by sweeping the wing. Although the booster will not necessarily fly at equilibrium at the high sweep angles, these results do indicate how the higher sweep angles are better suited to flight at higher velocities and altitude.

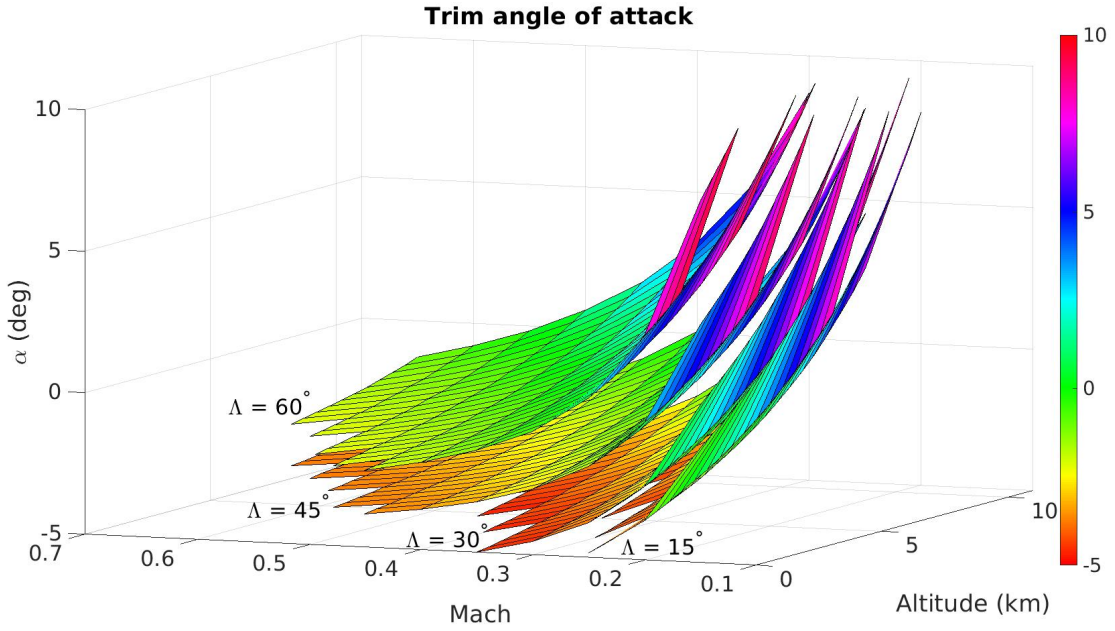


Figure 3.14: Trim angle of attack as a function of altitude and Mach number for wing sweep angles of 0°, 15°, 30° and 60°.

Algorithm 1 Newton-Raphson iteration to determine an equilibrium condition.

1: $\hat{\mathbf{x}}(0) = [0,0,0,0,0,0]^T$
2: $\epsilon = 10^{-10}$, $k = 0$
3: **while** $\|\mathbf{f}(\hat{\mathbf{x}}(k))\| > \epsilon$ **do**
4: $\quad \mathbf{J}(k) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Big|_{\hat{\mathbf{x}}(k)}$
5: $\quad \hat{\mathbf{x}}(k+1) = \hat{\mathbf{x}}(k) - \mathbf{J}^{-1}(k)\mathbf{f}(\hat{\mathbf{x}}(k))$
6: $\quad k++$
7: **end while**
8: **return** $\hat{\mathbf{x}}(k)$

Chapter 4

Vehicle Design Analysis

This chapter analyses the design of the first stage rocket booster in the various configurations: rocket, oblique-wing and aircraft. The result of this analysis provides constraints on the deployment of the wing and on the manoeuvre to begin the return flight, and is a significant contribution to the design of the ALV first stage boosters. We first discuss the addition of the elevons to the booster empennage, which increases the lateral disturbance rejection capabilities in the rocket configuration. We then analyse the critical Mach number, which defines the velocity at which the wing may be deployed, before defining the constraints of maneuvering flight in the oblique-wing and aircraft configurations. A pull-up manoeuvre is used to slow down and regain altitude after partially deploying the wing, and then a minimum sink rate glide is obtained to maximise the time available to start the piston engine.

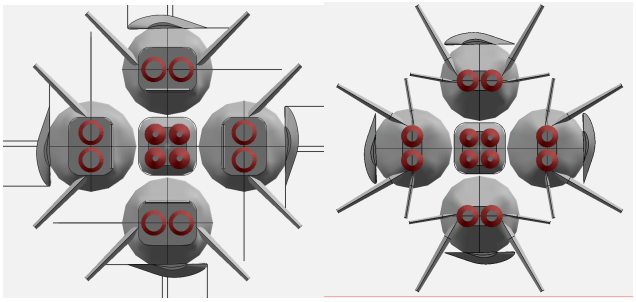
4.1 Empennage Design in the Rocket Configuration

This section discusses a modification made to the design of the first stage booster empennage, namely the addition of the elevons, which is one contribution made in this work. The elevons are introduced in order to provide additional lateral-directional control authority to better reject external disturbances, such as atmospheric winds.

4.1.1 Design Modification: Elevons

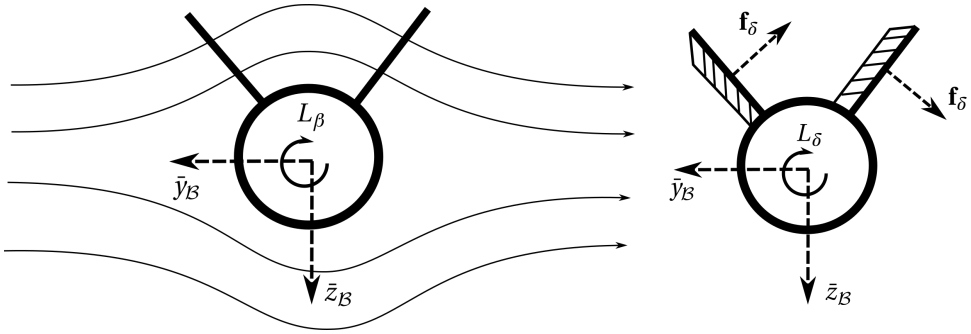
During the return flight in the rocket configuration, the fins on the empennage are the only actuators available to satisfy the control objectives. Therefore, the empennage control surfaces must track a longitudinal state reference in order to slow down, as well as stabilise lateral-directional motion. An early design of the first stage boosters used a V-tail empennage design. The motivation for selecting a V-tail design for the empennage results from the constraints of placing multiple first stage boosters in a parallel launch configuration around the centre upper stage rockets, as shown in Figure 4.1a. The empennage fins must be large enough to provide sufficient control authority throughout the reentry, but small enough such that they do not interfere with the fins from the other first stage boosters in the launch configuration. A V-tail

design is an attractive solution, as it provides both longitudinal and lateral-directional control and is not blocked by the stowed wing in the rocket configuration.



(a) (b)

Figure 4.1: Rear view of the ALV in the launch configuration (a) without elevons (b) with elevons.



(a) (b)

Figure 4.2: Front view of the first stage booster in the rocket configuration without the elevons: (a) under positive sideslip conditions, (b) rudder deflection required to zero sideslip.

However, the challenge with using the V-tail only during the return flight is that the system is under-actuated; there are two independent control surfaces to control motions about three axes. Specifically, the V-tail is unable to independently control yawing and rolling motions. Figure 4.2 exemplifies this by showing a front view of the booster under sideslip conditions. Airflow moving laterally over the fuselage interacts with the V-tail to produce a large rolling moment, causing a very large roll angle disturbance because of the low moment of inertia about the roll axis. If the V-tail deflects to produce a yawing moment to regulate sideslip and to align the booster with the airflow, an additional roll moment is produced in the same direction as the roll moment produced by the sideslip. Alternatively, if the V-tail deflects to counteract the roll moment resulting from the sideslip, a yaw moment will be produced that further increases the sideslip.

This challenge motivated the introduction of additional control surfaces in the rocket configuration to make the system over-actuated. Small fins on the empennage below the V-tail, henceforth called elevons, offer the ability to independently control rolling and yawing motions.

4.1.2 Design Comparison

To verify the design decision to include the elevons, we investigate the lateral-directional dynamics of the booster in the rocket configuration with and without the elevons, and design example feedback controllers for each configuration to regulate the roll angle. A detailed analysis of this exercise is provided in Appendix B, and the important insights and conclusions are presented here.

Natural Dynamics

The linearised dynamics for the booster are determined at an altitude of 9 km and at Mach 0.8 using the procedure in Section 3.5.1. The lateral-directional state variables are the sideslip angle β, the roll rate p, the yaw rate r and the roll angle Φ, and may be decoupled from the longitudinal states due to the near symmetry of the rocket configuration.

	Poles	Damping	Natural Frequency (rad/s)
Open loop without elevons	$p_{1,2} = -0.178 \pm 0.620i$ $p_{3,4} = -2.45 \pm 7.86i$	$\zeta_{1,2} = 0.275$ $\zeta_{3,4} = 0.298$	$\omega_{1,2} = 0.644$ $\omega_{3,4} = 8.22$
Open loop with elevons	$p_{1,2} = -0.289 \pm 0.576i$ $p_{3,4} = -2.44 \pm 7.91i$	$\zeta_{1,2} = 0.448$ $\zeta_{3,4} = 0.294$	$\omega_{1,2} = 0.644$ $\omega_{3,4} = 8.28$
Closed loop without elevons	$p_{1,2} = -0.911 \pm 0.934i$ $p_{3,4} = -1.69 \pm 6.24i$ $p_5 = -2.78$	$\zeta_{1,2} = 0.698$ $\zeta_{3,4} = 0.261$	$\omega_{1,2} = 1.30$ $\omega_{3,4} = 6.47$
Closed loop with elevons	$p_{1,2} = -0.755 \pm 0.758i$ $p_{3,4} = -14.2 \pm 24.3i$ $p_5 = -3.20$	$\zeta_{1,2} = 0.706$ $\zeta_{3,4} = 0.504$	$\omega_{1,2} = 1.07$ $\omega_{3,4} = 28.2$

Table 4.1: Poles, damping and natural frequency of the open loop and closed loop lateral directional dynamics with and without elevons.

Table 4.1 lists the open loop and closed loop poles of the lateral-directional dynamics with and without the elevons. There are two stable, oscillatory modes of motion, both of which have poor damping. One mode is very fast and characterises the weathercocking effect, and is similar to the Dutch Roll mode. Given a sideslip perturbation, the tail fins cause a yawing moment to align the booster with the airflow. However, due to the interaction of the airflow with the V-tail, a large transient roll angle results from the sideslip perturbation. The second mode is much slower, and is a combination of the roll and spiral modes. This mode describes the tendency of the booster to return to zero roll angle after a roll angle disturbance. With a roll angle, gravity induces a sideslip that produces a moment to counter the roll angle disturbance. Adding the elevons increases the damping of the slow roll mode, while having negligible effect on the weathercocking mode.

Lateral-Directional Controller

Feedback controllers, similar to common designs for small UAVs such as in [8] or [40], are designed for the booster with and without the elevons using the root locus design methodology. The primary objective of these controllers is to stabilise the roll angle in the presence of sideslip disturbances. Specifically, the controllers are designed to achieve a roll mode damping of 0.7 ($\zeta_{1,2} = 0.7$), while improving or maintaining the damping of the weathercocking mode. A block diagram for these controllers is shown in Figure 4.3. Inner-loops feed back the roll and yaw rates to increase damping, and an outer loop proportional-integral controller regulates the roll angle. In the original design, both δ_a and δ_r map to the V-tails; in the modified design, δ_a maps to the elevons, and δ_r maps to the V-tail. As shown in Table 4.1, the controller without the elevons achieves a critically damped roll mode, however reduces the damping of the weathercocking mode. Adding the elevons provides the capability to achieve a critically damped roll mode while also increasing the damping of the weathercocking mode.

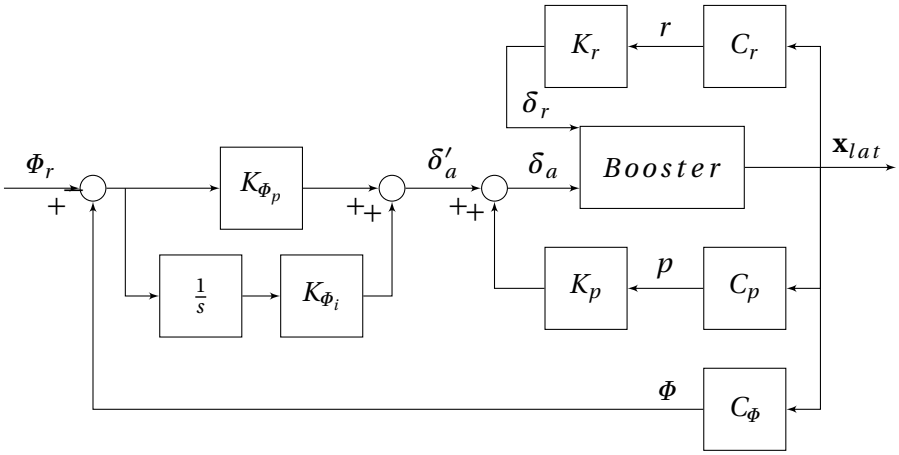


Figure 4.3: Block diagram for lateral-directional controller for the rocket configuration. Without the elevons, the V-tail is deflected for both δ_r and δ_a commands.

To verify the performance of the controller designs, the lateral-directional dynamics are simulated in Simulink with a 1° initial sideslip and zero roll angle, roll rate and yaw rate. Figure 4.4 shows the state responses for the open loop and closed loop system with and without the elevons. Figure 4.5 shows the closed loop control inputs with and without the elevons. For both open loop systems, the initial 1° sideslip angle causes an approximately −14° roll angle disturbance. The fast weathercocking mode causes the booster to quickly align with the airflow, however the slow roll mode is excited and takes approximately 10 seconds to settle. We note the small effect adding the elevons has on increasing the speed at which the roll mode settles in the open loop response. The benefit for the closed loop performance is however much more significant. Without the elevons, the lateral-directional controller is able to stabilise roll faster than the open loop response, however actually results in a larger overshoot of the sideslip angle. With the elevons, the controller achieves a critically damped response for the weathercocking mode while regulating the roll angle to a maximum transient roll angle of approximately −2°. This improved performance

results from the elevons immediately rejecting roll disturbances, leaving the V-tail to simply dampen yaw rates.

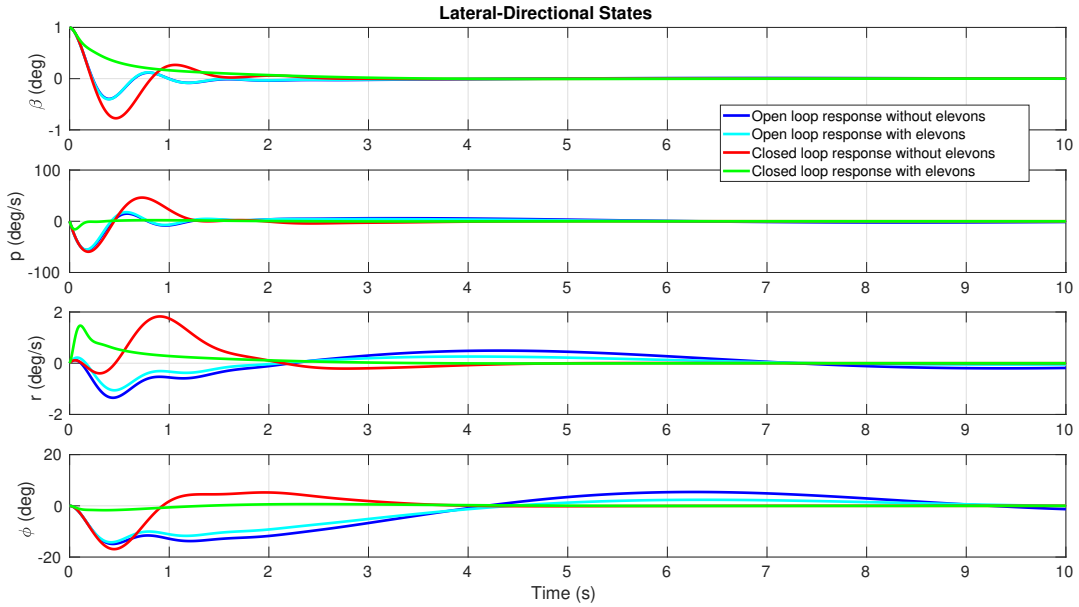


Figure 4.4: Open loop and closed loop responses of the lateral states to a 1° initial sideslip angle with and without the elevons.

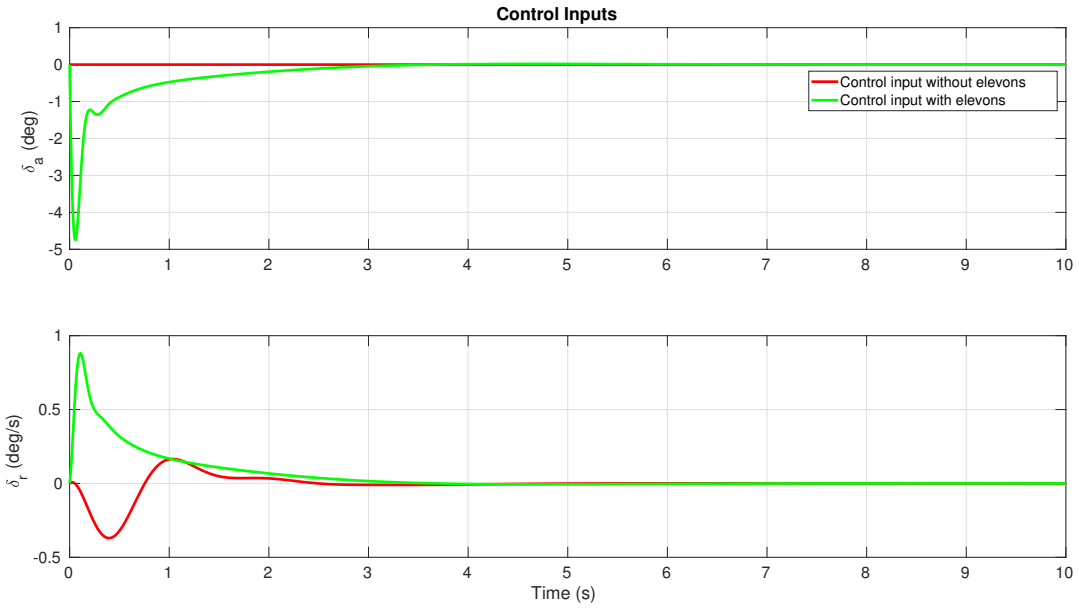


Figure 4.5: Closed loop control inputs to regulate the roll angle with a 1° initial sideslip angle with and without the elevons.

Note that since the sideslip angle is not being regulated in this design, and instead yaw rates are dampened, the resulting V-tail deflection to a sideslip perturbation will initially be opposite to that shown in Figure 4.2b. This would result in a roll moment that opposes the roll moment caused by the sideslip. As shown in Figure 4.4, the booster briefly experiences a positive yaw moment caused by the weathercocking

effect of the V-tail, which results from the stability derivative for the change in yaw moment due to sideslip, C_{n_β}. The corresponding V-tail deflection to dampen this yaw moment provides a restoring roll moment. However, an additional complication is that the booster experiences a very large roll rate due to the airflow interaction with the V-tail. This roll rate produces a yaw moment, quantified by C_{n_p}, that is caused by the V-tail surfaces seeing a differential angle of attack during rolling motions. This differential angle of attack results in a differential lift, and therefore drag, that causes a yaw moment in the same direction as the roll rate. This is observed in Figure 4.4, as the roll rate quickly becomes large resulting in the yaw moment contribution of C_{n_p} outweighing the yaw moment contribution of C_{n_β}. This causes the booster to first experience a positive yaw rate, then a negative yaw rate once the roll rate builds up. Damping the yaw rate once negative results in the deflection shown in Figure 4.2b, causing a roll moment in the same direction as that caused by the sideslip angle. This causes the roll moment disturbance due to a sideslip perturbation to be larger for the closed loop system with only the V-tail than for the open loop system.

A significant benefit of adding the elevons is that this large roll rate due to sideslip can be prevented without using the V-tail. As seen in Figure 4.5, the elevons quickly generate a roll moment to prevent the large roll rate, which prevents the yaw contribution of C_{n_p} becoming larger than that of C_{n_β} as seen in Figure 4.4. This allows the V-tail to then dampen the positive yaw rate resulting from the weathercocking effect, which further aids in the regulation of the roll angle.

4.2 Constraints on the Return Flight

This section discusses the constraints and objectives on deploying the wing and on the manoeuvre to begin the return flight. Overall, the objectives of the control system for the subsonic flight are to stabilise the booster throughout the wing deployment, and then to provide as much time as possible to start the piston engine for the return flight. In order to achieve this, it is desirable to first deploy the wing as early as possible in the flight. Then, with the wing fully deployed, the booster must obtain a minimum sink rate glide, which maximises the endurance and therefore the time the booster has to start the engine.

4.2.1 Wing Critical Mach Number

The primary constraint for deploying the wing is the wing's critical Mach number. The critical Mach number M_{cr} is the free-stream Mach number M_∞ at which airflow over the top surface of the airfoil reaches sonic speeds ($M = 1.0$) [37]. This causes a significant increase in drag, and will produce a shock wave if M_{cr} is exceeded, which may stall the wing. For swept wings, such as when the first stage booster wing is partially deployed, the critical Mach number increases since the airfoil sees a reduced component of the free stream airflow. Figure 4.6 shows how the sweep angle reduces the component of the airflow seen by the airfoil.

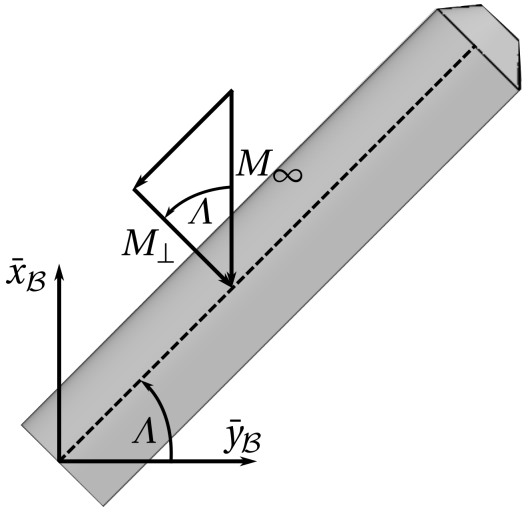


Figure 4.6: Right half of the first stage booster wing at a swept angle Λ.

For the ideal case of an infinite wing, the critical Mach number of a swept wing may be related to the zero sweep critical Mach number as [41]

$$M_{cr} = \frac{M_{cr,\Lambda=0}}{\cos\Lambda}. \tag{4.2.1}$$

Two methods are considered to estimate the critical Mach number of the booster's wing: computational fluid dynamic (CFD) analysis and compressibility corrections based on pressure coefficients obtained from PANUKL. The critical Mach number is determined for sweep angles of 0°, 15°, 30°, 45° and 60°.

Panel Method Analysis

As stated in Section 11.6 of [37], the location of the maximum airspeed over an airfoil corresponds to the lowest coefficient of pressure C_P, and does not vary significantly with Mach number in the subsonic regime. Furthermore, the pressure coefficient that results in airflow of Mach 1.0, denoted $C_{P,cr}$ is given as a function of M_∞ as

$$C_{P,cr} = \frac{2}{\kappa M_\infty^2}\left[\left(\frac{1+[(\kappa-1)/2]M_\infty^2}{1+[(\kappa-1)/2]}\right)^{\kappa/(\kappa-1)} - 1\right], \tag{4.2.2}$$

where $\kappa = 1.4$ is the ratio of specific heats for air. Equation 4.2.2 is a property of isentropic flow and thus independent of the airfoil. Since the location of the maximum airspeed on the airfoil does not vary with Mach number, we can use low-order analysis methods for incompressible flow, such as PANUKL, to first determine the location of the minimum pressure coefficient $C_{P,min}$ on the wing. Then, the Prandtl-Glauert compressibility correction, Equation 3.4.15, is applied to estimate the variation in $C_{P,min}$ with Mach number. The Mach number at which $C_{P,min}$ is equal to $C_{P,cr}$ is by definition the critical Mach number. This is most easily determined graphically, with Figure 4.7 showing the variation in $C_{P,min}$ with Mach number

for each of the considered wing sweep angles. The intersection of these curves with Equation 4.2.2 gives the critical Mach number for that sweep angle, with the results tabulated in Table 4.2.

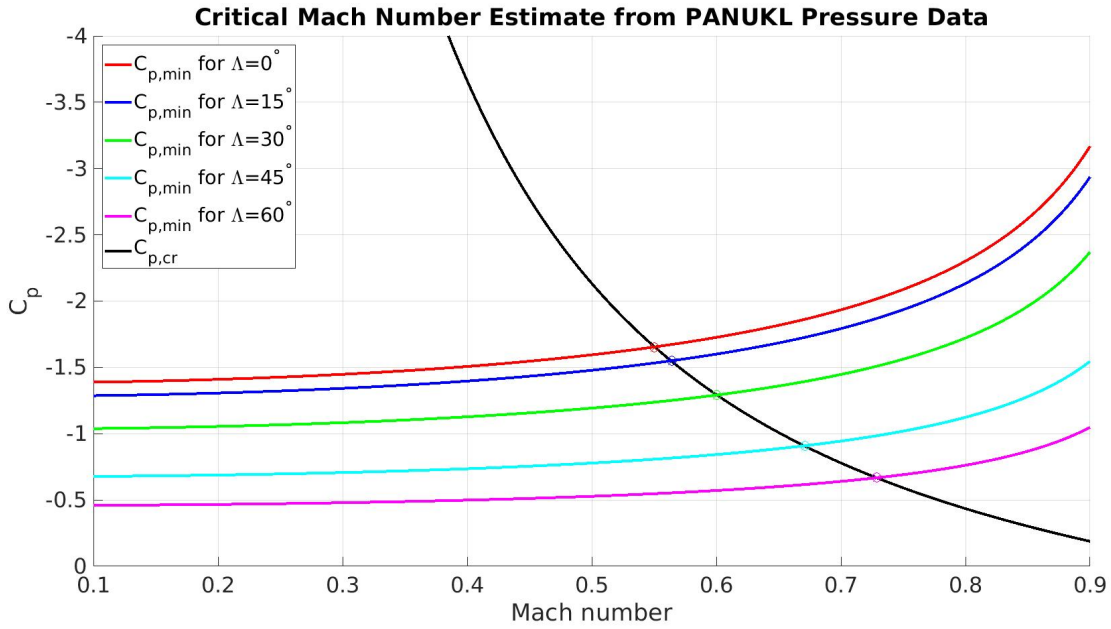


Figure 4.7: The pressure coefficients from PANUKL with a Prandtl-Glauert compressibility correction is shown by the colored lines for various wing sweep angles. The intersection of these curves with the black line, Equation 4.2.2, provides an estimate of the critical Mach number.

Λ	0°	15°	30°	45°	60°
M_{cr}	0.55	0.56	0.60	0.67	0.73

Table 4.2: Critical Mach number as a function of wing sweep angle from the PANUKL analysis.

Computational Fluid Dynamic Analysis

The Stanford University Unstructured (SU2) [42] CFD software is also used to determine estimates of the wing critical Mach number using an inviscid Euler solver. An unstructured mesh of the booster wing is constructed using Gmsh [43] for the five sweep angles considered. The free-stream Mach number in SU2 is varied until the maximum Mach number on the top surface of the airfoil reaches Mach 1.0, providing an estimate for the critical Mach number M_{cr}. Table 4.3 shows the resulting critical Mach numbers for the wing sweep angles considered.

Λ	0°	15°	30°	45°	60°
M_{cr}	0.52	0.55	0.59	0.66	0.70

Table 4.3: Critical Mach number as a function of wing sweep angle from the CFD analysis.

Wing Deployment Constraint

The estimates of the critical Mach number for the wing as a function of wing sweep angle is shown in Figure 4.8 for both methods. The results from PANUKL and SU2 agree fairly well for all wing sweep angles.

Furthermore, the theoretical variation in M_{cr} with sweep angle from Equation 4.2.1 is calculated from the estimates of M_{cr} at zero sweep for both methods. The theoretical estimate of M_{cr} agrees with the numerical results up to a sweep angle of 30°, however begins to differ considerably above 45°. This is likely partly due to the assumption in Equation 4.2.1 of an infinite wing being more erroneous at the higher sweep angles. Furthermore, as shown on the left side of Figure 4.9, the location of the maximum airspeed in both methods over the highly swept wing occurred near the right tip on the transition between the straight and tapered portions of the wing. The transition between these two regions is very abrupt, and designing a smoother transition may help to increase the critical Mach numbers for larger sweep angles.

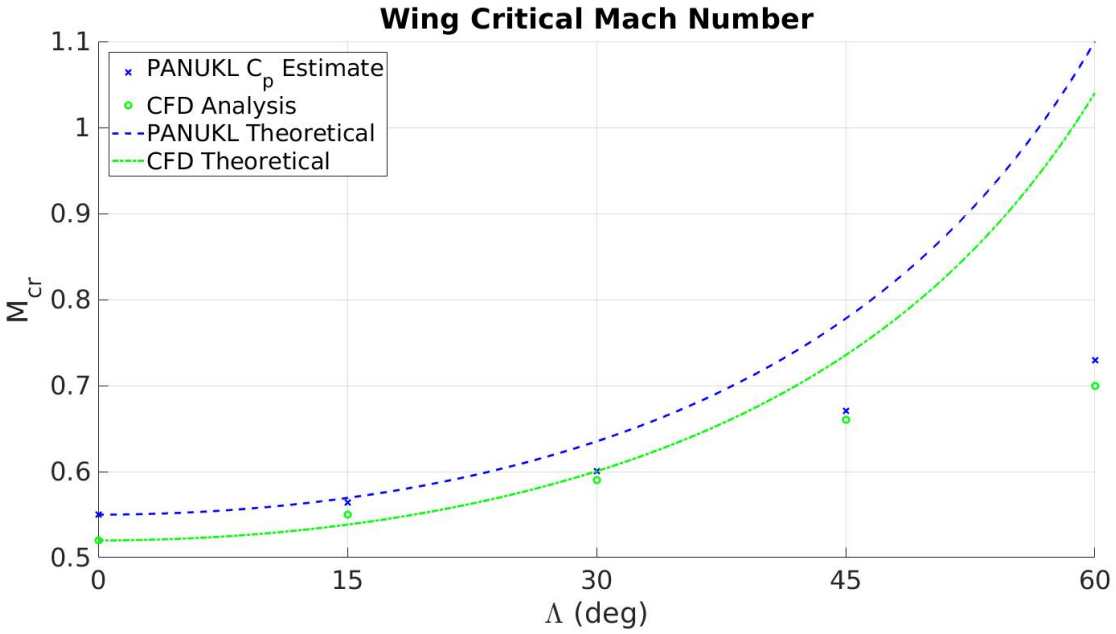


Figure 4.8: Wing critical Mach number estimates from PANUKL and SU2 as a function of the wing sweep angle.

However, optimising the shape of the wing is not in the scope of this work. Therefore it was decided to deploy the wing to 60° sweep at Mach 0.65, allowing for some margin of error in the preceding analysis. Figure 4.9 shows the results from the CFD analysis visualised in Paraview for a free-stream Mach number of 0.65 for wing sweep angles of 0° and 60°. Clearly, fully deploying the wing at this airspeed causes a significant region of supersonic flow to form, as shown by the red regions on the wing, while the flow remains subsonic over the wing at 60° sweep.

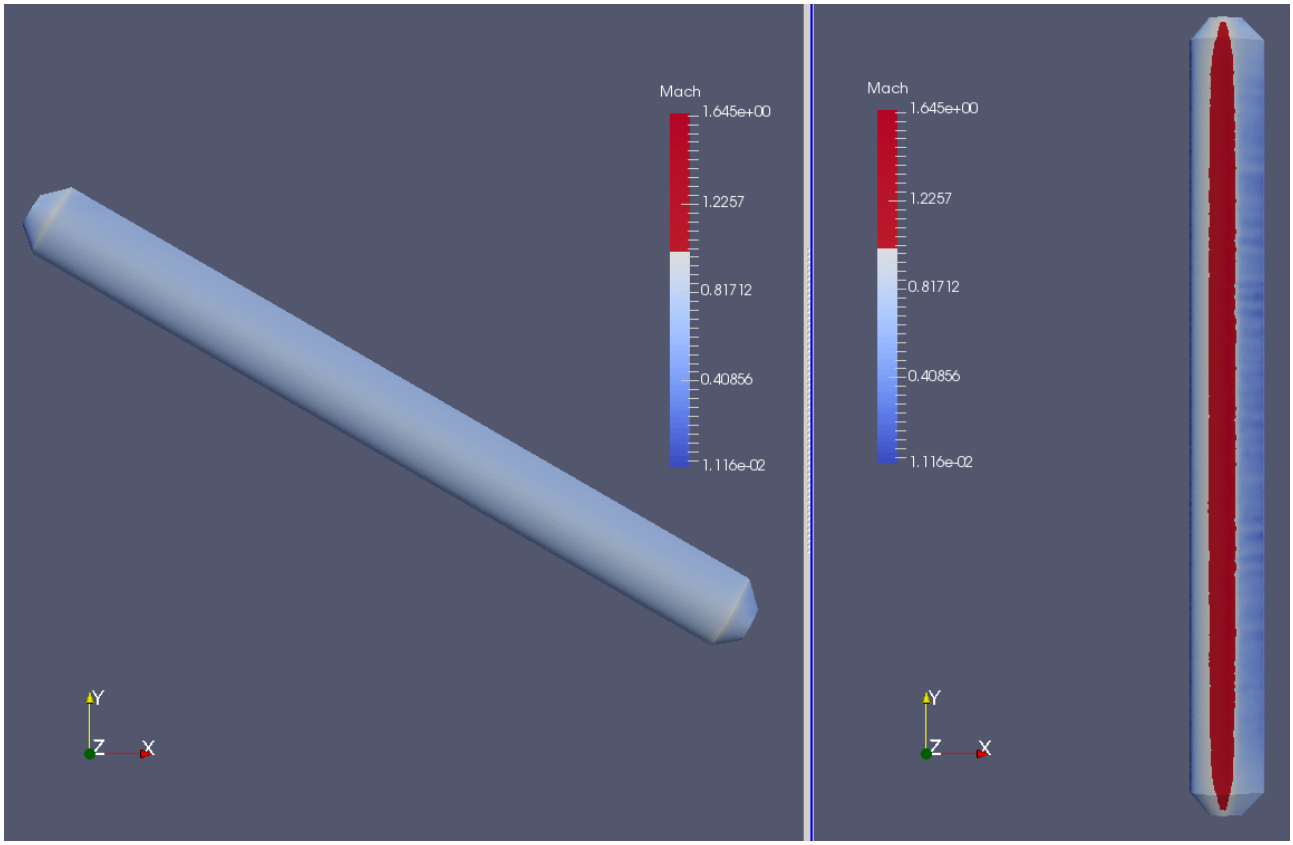


Figure 4.9: Computational fluid dynamics results for a wing sweep of 60° (left) and 0° (right) for a free-stream Mach number of 0.65. Regions on the wing colored red indicate supersonic flow

4.2.2 Minimum Sink Rate Glide

After deploying the wing and prior to starting the piston engine, the booster will essentially be a glider aircraft. In order to maximise the time available to start the piston engine, the booster must minimise the vertical velocity V_v, also called the sink rate. The sink rate for typical gliding flight is given as [44]

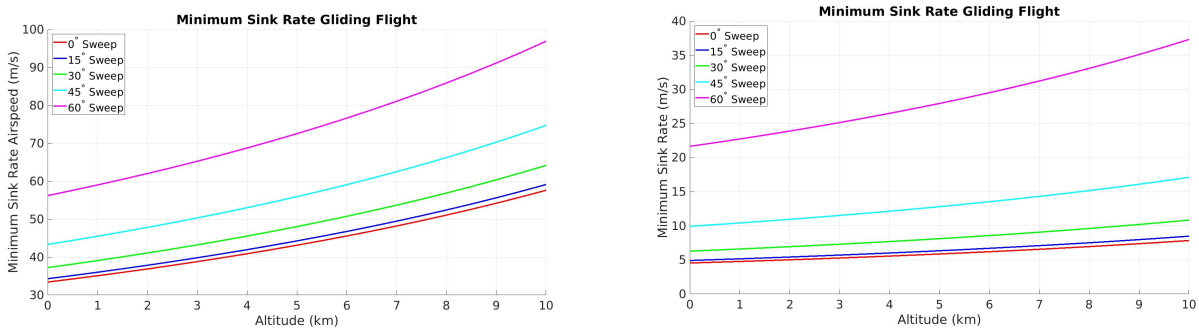
$$V_v = \frac{1}{\cos^{3/2}\Phi}\sqrt{\frac{W}{S}\frac{2}{\rho_\infty\left(C_L^3/C_D^2\right)}}, \tag{4.2.3}$$

where W is the booster weight, S is the wing planform area, ρ_∞ is the free-stream air density, C_L is the lift coefficient and C_D is the drag coefficient. The minimum sink rate is found for a given roll angle by maximising the term C_L^3/C_D^2. The airspeed at which this ratio is maximised is [44]

$$V_\infty = \sqrt{\frac{2W}{\rho_\infty S}\sqrt{\frac{K(\Lambda)}{3C_{D_0}}}}, \tag{4.2.4}$$

where $K(\Lambda) = \frac{1}{\pi \mathcal{AR}(\Lambda)e(\Lambda)}$ is the drag due to lift coefficient and C_{D_0} is the parasitic drag coefficient. In Equation 4.2.4, W, S and C_{D_0} are constant. Therefore, the minimum sink rate airspeed is only a function of the air density, which varies with altitude, and the drag due to lift coefficient, which varies with wing sweep angle. Figure 4.10a shows the airspeed required to obtain the minimum sink rate glide as a function of

altitude for wing sweep angles between 0° and 60°, and Figure 4.10b shows the resulting sink rate.



(a) (b)

Figure 4.10: Gliding flight as a function of altitude and wing sweep angle: (a) Airspeed required to maintain minimum sink rate glide (b) Resulting sink rate.

At the lower sweep angles, there is less induced drag and a larger lift coefficient, resulting in lower minimum sink rate airspeeds and lower sink rates. This is an intuitive result, as the wing produces more lift at 0° sweep, and implies that the wing should be fully deployed as soon as the booster is below the critical Mach number of the fully deployed wing in order to provide as much time as possible to start the piston engine.

4.2.3 Pull-up Manoeuvre

Once the wing has partially deployed to 60°, the booster must further decrease it's velocity to first fully deploy the wing, and then to obtain the minimum sink rate glide. Furthermore, there is a significant difference between the critical Mach number and the minimum sink rate airspeed for 0° sweep. Thus, it is desirable to store the excessive kinetic energy as gravitational potential energy to slow down, rather than allowing drag to dissipate this energy. Even in the oblique-wing configuration, the wing produces significant lift to allow for manoeuvring flight. Thus, the booster enters a pull-up manoeuvre at a sweep angle of 60°. Structural limitations constrain the rate at which the booster may pull up, and is quantified by the load factor. The load factor n is defined as the ratio of the lift force to the weight, and may be expressed in terms of the normal acceleration a_z in the body $\bar{z}_{\mathcal{B}}$ direction as [45]

$$n = \frac{-a_z}{g} + 1, \tag{4.2.5}$$

where g is the gravitational acceleration. Since the velocity is normal to the flight path, the normal acceleration is

$$a_z = -\dot{\gamma} V_\infty, \tag{4.2.6}$$

where γ is the flight path angle and V_∞ is the airspeed. Substituting Equation 4.2.6 into Equation 4.2.5 and gives

$$n = \frac{V_\infty}{g}\dot{\gamma} + 1. \tag{4.2.7}$$

Thus, we may limit the maximum load factor n_{max} during the pull-up manoeuvre by limiting the rate of change of the flight path angle as

$$\dot{\gamma} \leq \frac{g}{V_\infty}(n_{max} - 1). \tag{4.2.8}$$

4.2.4 Turn-around Manoeuvre

A secondary objective of the flight control system is to turn around in order to begin the return flight. To achieve this, the booster must obtain a heading angle reference χ provided by a guidance controller not considered in this work. From [44], the rate of change of the yaw angle for a glider aircraft in turning flight is

$$\dot{\Psi} = \frac{g}{V_\infty}\tan\Phi = \dot{\chi} = \frac{\Delta\chi}{\Delta t}. \tag{4.2.9}$$

Note that the heading and yaw angle may differ due to atmospheric winds, which will be treated as a disturbance by the control system. Therefore, a desired heading angle may be obtained by commanding a roll angle. However, recalling from Equation 4.2.3, the sink rate is inversely proportional to $\cos^{3/2}\Phi$. Thus, turning around to begin the return flight and obtaining the minimum sink rate glide are conflicting objectives. In order to achieve the objective of turning around with minimal impact on the time available to start the engine, we seek a roll angle that achieves a given change in heading $\Delta\chi$ with the smallest loss of altitude Δh. First, Equation 4.2.3 is re-written as

$$V_v = \frac{-\Delta h}{\Delta t} = \frac{1}{\cos^{3/2}\Phi}\sqrt{\frac{W}{S}\frac{2}{\rho_\infty\left(C_L^3/C_D^2\right)}} = \frac{1}{\cos^{3/2}\Phi}V_{v,0}, \tag{4.2.10}$$

where $-\Delta h$ is the altitude loss over a given time interval Δt, and $V_{v,0}$ is the sink rate at zero roll angle. Since the booster must turn around to fly back, $\Delta\chi$ will be a discrete and potentially large change that will take a significant Δt to achieve. Rearranging Equation 4.2.9 for Δt,substituting the result into 4.2.10 and solving for Δh gives

$$\Delta h = -\frac{\Delta t}{\cos^{3/2}\Phi}V_{v,0} = \Delta\chi\frac{V_\infty V_{v,0}}{g}\frac{1}{\sin\Phi\cos^{1/2}\Phi}. \tag{4.2.11}$$

Differentiating Equation 4.2.11 with respect to Φ and setting the result to zero gives

$$0 = \frac{\partial}{\partial \Phi}(\Delta h) = -\Delta\chi \frac{V_\infty V_{v,0}}{g}\left(\frac{1}{2\cos^{3/2}\Phi} - \frac{\cos^{1/2}\Phi}{\sin^2\Phi}\right) \tag{4.2.12}$$

Equation 4.2.12 can only be zero if

$$\begin{aligned} 0 &= \frac{1}{2\cos^{3/2}\Phi} - \frac{\cos^{1/2}\Phi}{\sin^2\Phi} = \frac{1}{2\cos^2\Phi} - \frac{1}{\sin^2\Phi} \\ &\Rightarrow \sin^2\Phi = 2\cos^2\Phi. \end{aligned} \tag{4.2.13}$$

Therefore, the roll angle that achieves a given change in heading with the smallest loss of altitude is

$$\Phi = \tan^{-1}\left(\sqrt{2}\right) = 54.7°. \tag{4.2.14}$$

However, this is a very large roll angle, and may be difficult to obtain without saturating the control surfaces at lower velocities. Since the turn around may be achieved faster, and therefore with less altitude loss, at the lower velocities, banking at 54.7° is not ideal. Figure 4.11 plots Equation 4.2.11 for 0° sweep on a minimum sink rate glide at an altitude of 5 km for a heading angle change of 180°, and suggests that banking at as large of an angle as possible up to 54.7° provides a reduction in the altitude loss resulting from a change in heading.

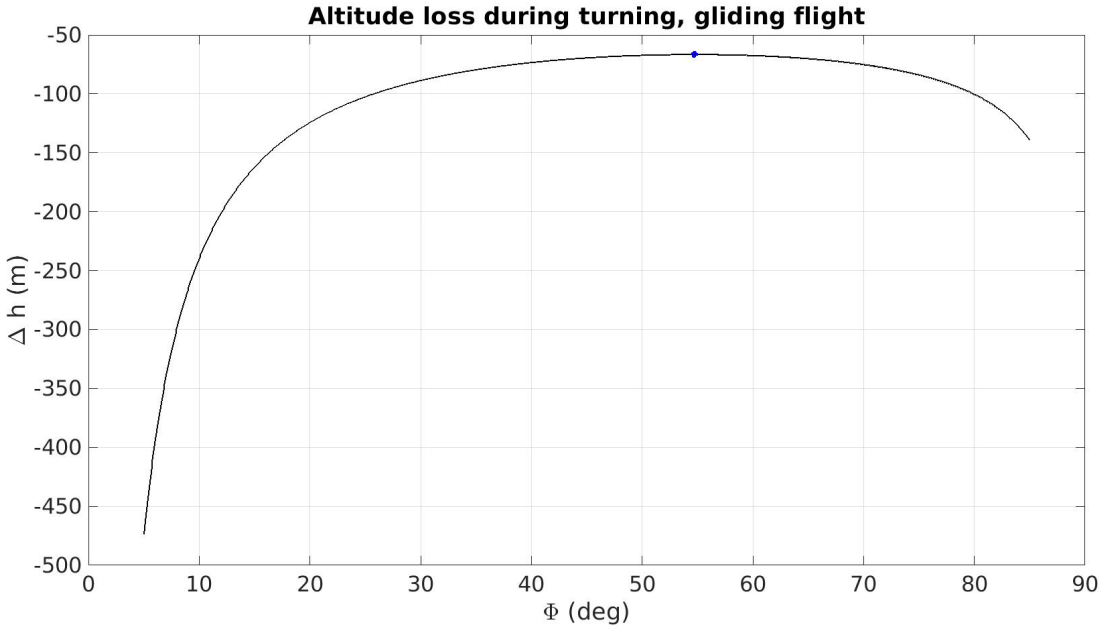


Figure 4.11: Altitude loss as a function of the roll angle required to achieve a change in heading of 180° during turning, gliding flight at 5 km altitude.

This analysis implies that turning around to begin the fly back phase of the return flight has a significant, negative impact on the ability to achieve the primary objective of maximising the time available to

start the piston engine. Therefore, once the minimum sink rate glide has been obtained, the booster will instead maintain the original heading prior to deploying the wing. The turn around manoeuvre will then be performed after starting the piston engine and deploying the propellers.

4.3 Conclusion

This chapter showed that adding the elevons significantly improves lateral-directional controllability of the booster in the rocket configuration. Furthermore, a concept of operations for the deployment of the wing has been defined. The wing first deploys to a sweep angle of 60° at Mach 0.65, which is less than the estimated critical Mach number at 60° but greater than the critical Mach number at 0°. Furthermore, the minimum sink rate glide resulting in the smallest sink rate occurs at a sweep angle of 0°, as the wing produces the most lift when fully deployed. Thus, the booster performs a pull-up manoeuvre after first partially deploying the wing, allowing the booster to slow down by exchanging kinetic energy for gravitational potential energy. The rate at which the booster may pull-up is limited by the load factor. The wing is then fully deployed to 0° sweep and the minimum sink rate glide is obtained after the pull up manoeuvre.

This chapter also showed that the turn around manoeuvre to begin the return flight conflicts with the primary requirement of obtaining the minimum sink rate glide to maximise the time available to start the piston engine. Furthermore, the bank angle of 54.7° that minimises the loss in altitude required to perform a change of heading is very large. Therefore, the booster will instead maintain a constant heading angle prior to starting the engine, and only initiate the turn-around manoeuvre after deploying the propellers. The design of these constraints for the deployment of the wing is a significant contribution of this work, and motivates the controller design presented in the following chapter.

Chapter 5

Control System Design

This chapter develops the control system for the wing deployment of the first stage booster. The primary objectives are to stabilise the booster in the subsonic rocket configuration, to stabilise the attitude throughout the deployment of the wing and to obtain the minimum sink rate glide once the wing has fully deployed. This chapter forms the primary contribution of this research: a flight control system that is capable of handling the significant variation in the booster's configuration resulting from the deployment of the wing.

5.1 Multivariate Control Theory

The linearised dynamics from Chapter 3 are

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\boldsymbol{\delta}, \tag{5.1.1}$$

where the state is

$$\mathbf{x} = \begin{bmatrix} u/V_\infty & \beta & \alpha & P & Q & R & \Phi & \Theta \end{bmatrix}^T, \tag{5.1.2}$$

and the control input is

$$\boldsymbol{\delta} = \begin{bmatrix} \delta_{a,r} & \delta_{a,l} & \delta_{e,r} & \delta_{e,l} & \delta_{r,r} & \delta_{r,l} \end{bmatrix}^T. \tag{5.1.3}$$

Since the dynamics are highly coupled during the oblique wing case, it is not obvious how to design cascaded single-input, single-output (SISO) controllers as is typical for aircraft flight control systems. Therefore, a multiple-input, multiple-output (MIMO) control architecture is designed using Linear Quadratic Regulator (LQR) theory. The rest of this section derives the LQR feedback gain for a MIMO system.

5.1.1 Linear Quadratic Regulator

Since the controllers will be implemented on a microprocessor, the dynamic system from Equation 5.1.1 must first be discretised. Using a zero-order hold with a sampling time of T_s, the discrete dynamics are

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \boldsymbol{\delta}_k, \tag{5.1.4}$$

where

$$\mathbf{A}_k = e^{\mathbf{A}T_s} \tag{5.1.5}$$

and

$$\mathbf{B}_k = \int_0^{T_s} e^{\mathbf{A}\tau}\,\mathbf{B}\,d\tau. \tag{5.1.6}$$

Since the plant is slowly time varying, the state and input coefficient matrices are assumed to be constant and simply denoted as $\mathbf{A}$ and $\mathbf{B}$, respectively. LQR seeks to regulate the state of a dynamic system while also considering the control authority required. The following derivation of the LQR feedback control law is derived in [46], and presented here for completeness. An infinite-horizon cost function is defined as

$$J = \frac{1}{2}\sum_{k=0}^{\infty}\left(\mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \boldsymbol{\delta}_k^T \mathbf{R}\boldsymbol{\delta}_k\right), \tag{5.1.7}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are weighting matrices used to tune the performance of the controller by balancing control authority power with speed of regulation. To determine a control law that drives the state to zero, the Hamiltonian is first constructed as

$$H_k = \frac{1}{2}\left(\mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \boldsymbol{\delta}_k^T \mathbf{R}\boldsymbol{\delta}_k\right) + \boldsymbol{\xi}_{k+1}^T(\mathbf{A}\mathbf{x}_k + \mathbf{B}\boldsymbol{\delta}_k), \tag{5.1.8}$$

where $\boldsymbol{\xi}_{k+1}^T$ is an unknown Lagrange multiplier. Differentiating 5.1.8 with respect to $\boldsymbol{\delta}_k$ and setting to zero gives

$$\frac{\partial H_k}{\partial \boldsymbol{\delta}_k} = \mathbf{R}\boldsymbol{\delta}_k + \mathbf{B}^T\boldsymbol{\xi}_{k+1} = 0, \tag{5.1.9}$$

and $\boldsymbol{\delta}_k$ is solved for as

$$\boldsymbol{\delta}_k = -\mathbf{R}^{-1}\mathbf{B}^T\boldsymbol{\xi}_{k+1}. \tag{5.1.10}$$

The state and costate equations are found by differentiating Equation 5.1.8 with respect to $\boldsymbol{\xi}_{k+1}$ and $\mathbf{x}_k$, respectively, giving

$$\mathbf{x}_{k+1} = \frac{\partial H_k}{\partial \boldsymbol{\xi}_{k+1}} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\boldsymbol{\delta}_k \tag{5.1.11}$$

and

$$\boldsymbol{\xi}_k = \frac{\partial H_k}{\partial \mathbf{x}_k} = \mathbf{Q}\mathbf{x}_k + \mathbf{A}^T\boldsymbol{\xi}_{k+1}. \tag{5.1.12}$$

Substituting Equation 5.1.10 into the state equation gives

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k - \mathbf{B}\,\mathbf{R}^{-1}\mathbf{B}^T\boldsymbol{\xi}_{k+1}. \tag{5.1.13}$$

We now assume that $\boldsymbol{\xi}_k$ and $\mathbf{x}_k$ are related by an unknown matrix $\mathbf{S}_k$ as

$$\boldsymbol{\xi}_k = \mathbf{S}_k\mathbf{x}_k, \tag{5.1.14}$$

so that Equation 5.1.13 may be expressed as

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k - \mathbf{B}\,\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}_{k+1}\mathbf{x}_{k+1}. \tag{5.1.15}$$

Solving Equation 5.1.15 for $\mathbf{x}_{k+1}$ results in

$$\mathbf{x}_{k+1} = \left(\mathbf{1} + \mathbf{B}\,\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}_{k+1}\right)^{-1}\mathbf{A}\mathbf{x}_k, \tag{5.1.16}$$

which is substituted in to the costate equation giving

$$\boldsymbol{\xi}_k = \mathbf{S}_k\mathbf{x}_k = \mathbf{Q}\mathbf{x}_k + \mathbf{A}^T\mathbf{S}_{k+1}\left(\mathbf{1} + \mathbf{B}\,\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}_{k+1}\right)^{-1}\mathbf{A}\mathbf{x}_k. \tag{5.1.17}$$

Equation 5.1.17 must hold for all $\mathbf{x}_k$, thus

$$\mathbf{S}_k = \mathbf{Q} + \mathbf{A}^T\mathbf{S}_{k+1}\left(\mathbf{1} + \mathbf{B}\,\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}_{k+1}\right)^{-1}\mathbf{A}. \tag{5.1.18}$$

By assuming that $\mathbf{S}_k$ converges as $k \to \infty$, we can replace $\mathbf{S}_k$ and $\mathbf{S}_{k+1}$ with a constant $\mathbf{S}$, which is known as the algabraic Ricatti equation. There are software implementations to solve the algabraic Ricatti equation, and this work using MATLAB's *lqr.m*. The control input is then

$$\boldsymbol{\delta}_k = -\mathbf{R}^{-1}\mathbf{B}^T\boldsymbol{\xi}_{k+1} = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\mathbf{x}_{k+1}. \tag{5.1.19}$$

This is highly undesirable since the control input at the current time step relies on knowing the state at the next time step. We can substitute the state equation into Equation 5.1.19 to give

$$\boldsymbol{\delta}_k = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\left(\mathbf{A}\mathbf{x}_k + \mathbf{B}\boldsymbol{\delta}_k\right), \tag{5.1.20}$$

which, after pre-multiplying by $\mathbf{R}$, is solved for $\boldsymbol{\delta}_k$ as

$$\boldsymbol{\delta}_k = -\left(\mathbf{R} + \mathbf{B}^T\mathbf{S}\,\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{S}\mathbf{A}\mathbf{x}_k = -\mathbf{K}\mathbf{x}_k. \tag{5.1.21}$$

The feedback gain $\mathbf{K}$ may then be computed prior to implementation and then stored in memory.

5.1.2 Linear Quadratic Servo

Equation 5.1.21 provides a gain $\mathbf{K}$ that regulates the state $\mathbf{x}$ to zero as $k \to \infty$. However, we would like to pull up once the wing deploys to obtain the minimum sink rate glide. Thus, we would like to track a reference input rather than regulate the state to the linearisation point. One method for LQR tracking, also called Linear Quadratic Servo (LQS) is to modify the cost function to minimise the difference between the states to be tracked, obtained by the output matrix $\mathbf{C}$, and the reference input $\mathbf{r}_k$ [46] as

$$J = \frac{1}{2}\sum_{k=0}^{\infty}\left[(\mathbf{C}\mathbf{x}_k - \mathbf{r}_k)^T\,\mathbf{Q}\,(\mathbf{C}\mathbf{x}_k - \mathbf{r}_k) + \boldsymbol{\delta}_k^T\,\mathbf{R}\,\boldsymbol{\delta}_k\right]. \tag{5.1.22}$$

The feedback control law is then given as

$$\boldsymbol{\delta}_k = -\mathbf{K}\mathbf{x}_k - \mathbf{L}\mathbf{r}_k, \tag{5.1.23}$$

and the derivation is presented in Appendix C. Figure 5.1 shows the block diagram for this LQR tracking controller. The reference input is multiplied by a feedforward gain $\mathbf{L}$, and is therefore unable to sufficiently reject disturbances since the state is never compared to the reference in the feedback.

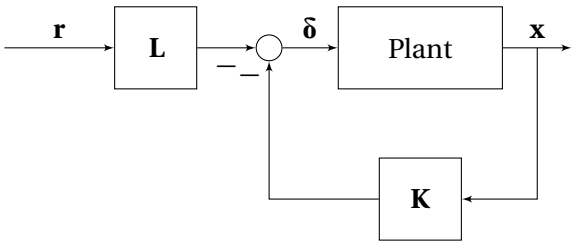


Figure 5.1: Block diagram for LQS with error between reference input and output state in the cost function.

Another method to use LQR design for a tracking controller is to design a typical LQR controller, but apply the gain to the error between the reference and the states to be tracked [47]. In this method, the full

state is defined as

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_y \\ \mathbf{x}_r \end{bmatrix}, \tag{5.1.24}$$

where $\mathbf{x}_y$ are the states selected by the output matrix $\mathbf{C}$, and $\mathbf{x}_r$ are the remaining states. Thus, the reference $\mathbf{r}$ must be a subset of the full state vector. The LQR gain is then determined as detailed in Section 5.1, providing the gain

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_y & \mathbf{K}_r \end{bmatrix}. \tag{5.1.25}$$

In this case the block diagram is shown in Figure 5.2.

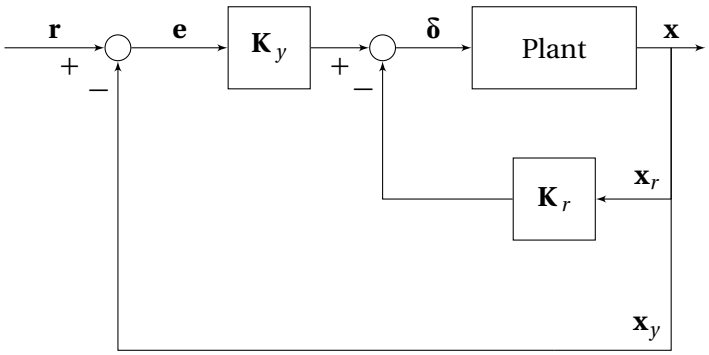


Figure 5.2: Block diagram for LQS with error between reference input and output state in the cost function.

This method of LQS allows the reference to be compared to the output state by design, however it still requires feedback of the remaining states. Since this work focuses on the subsonic flight of the boosters prior to starting the aero engine, the available control effectors are the aerodynamic control surfaces that primarily cause moments about the three body axes. Therefore, a natural choice for the reference states are the attitude or angular velocity states. This would however leave the airspeed as one of the states in $\mathbf{x}_r$ that will be regulated to the linearisation point, which contradicts with the objective of the pull up manoeuvre to reduce the airspeed.

Furthermore, multivariate systems can only achieve perfect tracking when the dimension of the control inputs is the same as the states to be tracked [48]. Again, the control surfaces primarily cause moments about the three body axes, reinforcing the choice of attitude or angular velocity states for the reference feedback. These insights motivated dividing the full state into multiple loops for feedback, similar to the cascaded control architectures that are common in UAVs. An inner-loop LQS controller tracks an angular velocity reference by commanding a moment from the control surfaces. An outer-loop LQS controller then tracks an attitude reference by commanding an angular velocity from the inner-loop controller. With the attitude controlled, external forces may then be used to control the velocity using typical SISO controllers. The resulting controller diagram is shown in Figure 5.3.

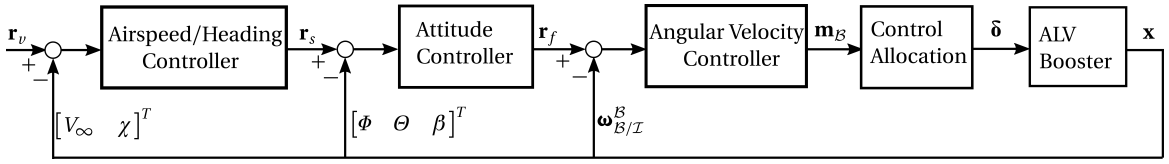


Figure 5.3: ALV First Stage Booster control architecture for the wing deployment.

5.2 Multiple-Loop, Multiple-Input, Multiple-Output

Figure 5.4 shows the block diagram for the two inner loops designed using the LQS design motivated by [47].

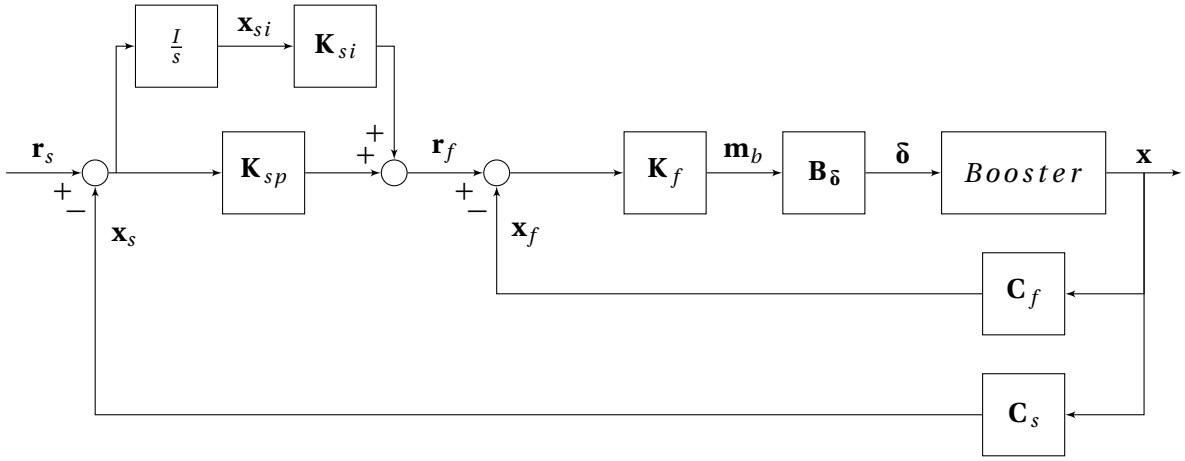


Figure 5.4: Block diagram for the multi-loop MIMO controller designed for the first stage booster.

The following sections develop the controllers for each loop, beginning with the very inner loop and moving outwards.

5.2.1 Control Allocation

The first stage booster has six independent control surfaces that may be combined to produce moments about the three-body axes, and therefore is over-actuated. Furthermore, the availability and effectiveness of these control surfaces varies considerably throughout the wing deployment. Control allocation is utilised to ease the burden of designing the outer-loop controllers by presenting a common interface to the outer-loop controllers that is independent of the current booster configuration. Since the available control effectors primarily produce moments, a control allocation algorithm is developed to distribute the available control effectors,

$$\boldsymbol{\delta} = \begin{bmatrix} \delta_{A,R} & \delta_{A,L} & \delta_{E,R} & \delta_{E,L} & \delta_{R,R} & \delta_{R,L} \end{bmatrix}, \tag{5.2.1}$$

to achieve a reference body moment $\mathbf{m}_{b,r}$ provided by the angular velocity controller. As described in [49], the control allocation problem may be formulated as determining the control input $\boldsymbol{\delta}$ such that

$$\mathbf{m}_{b,r} = \mathbf{B}_{\boldsymbol{\delta}}\boldsymbol{\delta}, \tag{5.2.2}$$

where $\mathbf{B}_{\boldsymbol{\delta}}$ is the control effectiveness matrix given as

$$\mathbf{B}_{\boldsymbol{\delta}} = \frac{\partial \mathbf{m}_b}{\partial \boldsymbol{\delta}}. \tag{5.2.3}$$

Thus, the control effectiveness matrix is found by differentiating external body moment from Equation 3.4.6, and it is easily observed that the result is simply the control derivatives for each of the individual control surfaces. Thus

$$\mathbf{B}_{\boldsymbol{\delta}} = Q_\infty S \begin{bmatrix} C_{l_{a,r}} & C_{l_{a,l}} & C_{l_{e,r}} & C_{l_{e,l}} & C_{l_{r,r}} & C_{l_{r,l}} \\ C_{m_{a,r}} & C_{m_{a,l}} & C_{m_{e,r}} & C_{m_{e,l}} & C_{m_{r,r}} & C_{m_{r,l}} \\ C_{n_{a,r}} & C_{n_{a,l}} & C_{n_{e,r}} & C_{n_{e,l}} & C_{n_{r,r}} & C_{n_{r,l}} \end{bmatrix}. \tag{5.2.4}$$

Furthermore, the solution to the control allocation problem must respect the saturation limits of the individual control surfaces given by

$$\boldsymbol{\delta}_{min} \leq \boldsymbol{\delta} \leq \boldsymbol{\delta}_{max}, \tag{5.2.5}$$

where $\boldsymbol{\delta}_{min}$ is a vector of the largest negative deflection of each surface and $\boldsymbol{\delta}_{max}$ is the largest positive deflection of each surface. During the initial wing deployment, the high sweep angle of the wing results in significant roll instability, so we would like to begin using the ailerons as early as possible. It is expected that the ailerons will quickly saturate during the wing deployment, and therefore any remaining control moment should be distributed among the remaining control surfaces.

Explicit Ganging

The simplest method to allocate the control surfaces is to use explicit ganging to manually assign the individual control effectors to groups for use in the outer loop controllers. In explicit ganging, the redundant control effectors are grouped into pseudo control effectors $\boldsymbol{\delta}_P$ by the control allocation matrix $\mathbf{G}$ as [49]

$$\boldsymbol{\delta} = \mathbf{G}\boldsymbol{\delta}_P. \tag{5.2.6}$$

Here the traditional aileron, elevator and rudder deflections are chosen for the pseudo control effectors, thus

$$\boldsymbol{\delta}_P = \begin{bmatrix} \delta_{a,P} \\ \delta_{e,P} \\ \delta_{r,P} \end{bmatrix}. \tag{5.2.7}$$

The traditional sign convention for aircraft control surface deflections is used for the pseudo effectors, in which a positive deflection of the pseudo effectors causes a negative moment. The convention in the rest of this thesis for the physical effectors is still used, in which a positive deflection downwards in the positive $\bar{z}_{\mathcal{B}}$ direction. In the rocket configuration, the ailerons on the wing are unavailable, so the elevons are used for pseudo aileron control. The ruddervators are used for rudder control and both the elevons and ruddervators are used for elevator control. Thus, the control allocation matrix for the rocket configuration is

$$\mathbf{G}_{rocket} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 1 & 1 \end{bmatrix}. \tag{5.2.8}$$

Once the wing deploys and the ailerons become available, the control allocation matrix for the aircraft configuration is

$$\mathbf{G}_{aircraft} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 1 & 1 \end{bmatrix}. \tag{5.2.9}$$

In order to provide a smooth transition between the rocket and aircraft configuration, the elements of $\mathbf{G}$ mapping pseudo aileron control to the physical control effectors are linearly varied over five seconds. Substituting Equation 5.2.7 into Equation 5.2.2 gives the reference moment as a function of the pseudo control effectors as

$$\mathbf{m}_{b,r} = \mathbf{B}_{\boldsymbol{\delta}} \mathbf{G} \boldsymbol{\delta}_P. \tag{5.2.10}$$

Noting that $\mathbf{B}_{\boldsymbol{\delta}}\mathbf{G}$ is a square matrix, $\boldsymbol{\delta}_P$ is found as

$$\boldsymbol{\delta}_P = (\mathbf{B}_{\boldsymbol{\delta}}\mathbf{G})^{-1}\mathbf{m}_{b,r}. \tag{5.2.11}$$

Substituting Equation 5.2.11 back into Equation 5.2.6 gives the control law as

$$\boldsymbol{\delta} = \mathbf{G}(\mathbf{B}_{\boldsymbol{\delta}}\mathbf{G})^{-1}\mathbf{m}_{b,r}. \tag{5.2.12}$$

While explicit ganging is simple and intuitive, it does not take into account the varying effectiveness of each individual control surface, and does not specifically consider the saturation limits. Furthermore, it is not able to account for saturated control surfaces. Explicit ganging is however useful as a baseline to compare to, and is also used in the determination of the equilibrium condition in Section 3.5.1.

Redistributed Pseudo Inverse

During the transition from the rocket to aircraft configuration, we would like to begin using the wing ailerons as soon as possible, and while the wing is initially rotating. At the very high sweep angles, the aileron will not be very effective and thus likely will not be able to provide the full roll moment required by $\mathbf{m}_{b,r}$, especially in the presence of disturbances. In this case, the remaining moment must be distributed among the remaining control surfaces. The pseudo inverse provides the minimum 2-norm to determine the control input $\boldsymbol{\delta}$ that solves the control allocation problem [50]. Here we present a process for deriving the pseudo inverse solution that respects the saturation limits of a control surface as described by Oppenheimer *et al.* [49]: the redistributed pseudo inverse. The approach presented here differs from the derivation in [49] in that we assume that the some of the control surfaces are saturated from the beginning, and desire to find the minimum 2-norm of the *unsaturated* control effectors, whereas Oppenheimer *et al.* derived the solution for an offset of the control effectors.

To handle the saturation of the control effectors, we consider $\boldsymbol{\delta}$ in terms of the unsaturated control effectors $\mathbf{u}$ and the saturated control effectors $\mathbf{s}$ as

$$\boldsymbol{\delta} = \mathbf{u} + \mathbf{s}. \tag{5.2.13}$$

When one of the control effectors saturate, the corresponding element of $\mathbf{u}$ is set to zero and the value of the deflection (i.e. $\boldsymbol{\delta}_{max}$ or $\boldsymbol{\delta}_{min}$) is placed into the corresponding element of $\mathbf{s}$. This implies that

$$\mathbf{u}\cdot\mathbf{s} = 0 \quad \forall \quad \boldsymbol{\delta}. \tag{5.2.14}$$

Substituting Equation 5.2.13 into Equation 5.2.2 gives

$$\mathbf{m}_{b,r} = \mathbf{B}_{\boldsymbol{\delta}}(\mathbf{u} + \mathbf{s}). \tag{5.2.15}$$

Since $\mathbf{u}\cdot\mathbf{s}=0$, $\mathbf{B}_{\boldsymbol{\delta}}$ may be expanded as

$$\mathbf{B}_{\boldsymbol{\delta}}=\mathbf{B}_{\mathbf{u}}+\mathbf{B}_{\mathbf{s}}, \tag{5.2.16}$$

where the matrices $\mathbf{B}_{\mathbf{u}}$ and $\mathbf{B}_{\mathbf{s}}$ contain the columns of $\mathbf{B}_{\boldsymbol{\delta}}$ corresponding to the non-zero elements of $\mathbf{u}$ and $\mathbf{s}$, respectively, and zeros elsewhere. This implies that $\mathbf{u}$ and $\mathbf{s}$ lie in the null spaces of $\mathbf{B}_{\mathbf{s}}$ and $\mathbf{B}_{\mathbf{u}}$, respectively. Therefore

$$\mathbf{B}_{\mathbf{s}}\mathbf{u}=0, \tag{5.2.17}$$

and

$$\mathbf{B}_{\mathbf{u}}\mathbf{s}=0. \tag{5.2.18}$$

Solving Equation 5.2.13 for $\mathbf{u}$ and substituting into Equation 5.2.15 gives

$$\begin{aligned}\mathbf{m}_{b,r}&=\mathbf{B}_{\mathbf{u}}(\boldsymbol{\delta}-\mathbf{s})+\mathbf{B}_{\mathbf{s}}\mathbf{s}\\&=\mathbf{B}_{\mathbf{u}}\boldsymbol{\delta}+\mathbf{B}_{\mathbf{s}}\mathbf{s}.\end{aligned} \tag{5.2.19}$$

When a control effector saturates, we desire to achieve the reference moment while minimising the deflection of the remaining unsaturated control surfaces. Thus, the cost function is defined as

$$J=\frac{1}{2}\mathbf{u}^T\mathbf{W}\mathbf{u}=\frac{1}{2}(\boldsymbol{\delta}-\mathbf{s})^T\mathbf{W}(\boldsymbol{\delta}-\mathbf{s}), \tag{5.2.20}$$

where $\mathbf{W}$ is a symmetric weighting matrix. The formulation of $\mathbf{B}_{\boldsymbol{\delta}}$ as the sum of $\mathbf{B}_{\mathbf{u}}$ and $\mathbf{B}_{\mathbf{s}}$, and the definition of the cost function in terms of $\mathbf{u}$ is the significant difference between the solution presented and that shown in [49]. We then select the Hamiltonian as

$$H=\frac{1}{2}\left(\mathbf{u}^T\mathbf{W}\mathbf{u}\right)+\boldsymbol{\xi}^T\left(\mathbf{B}_{\boldsymbol{\delta}}\boldsymbol{\delta}-\mathbf{m}_{b,r}\right), \tag{5.2.21}$$

where $\boldsymbol{\xi}$ is an unknown Lagrange multiplier to be determined. Substituting Equation 5.2.13 into Equation 5.2.21 and expanding the $\mathbf{B}_{\boldsymbol{\delta}}\boldsymbol{\delta}$ term as in Equation 5.2.19 gives

$$H=\frac{1}{2}\left(\boldsymbol{\delta}^T\mathbf{W}\boldsymbol{\delta}-\boldsymbol{\delta}^T\mathbf{W}\mathbf{s}-\mathbf{s}^T\mathbf{W}\boldsymbol{\delta}+\mathbf{s}^T\mathbf{W}\mathbf{s}\right)+\boldsymbol{\xi}\left(\mathbf{B}_{\mathbf{u}}\boldsymbol{\delta}+\mathbf{B}_{\mathbf{s}}\mathbf{s}-\mathbf{m}_{b,r}\right). \tag{5.2.22}$$

Taking the derivative of Equation 5.2.22 with respect to $\boldsymbol{\delta}$ and $\boldsymbol{\xi}$ and setting the results to zero gives

$$\frac{\partial H}{\partial\boldsymbol{\delta}}=\mathbf{W}\boldsymbol{\delta}-\frac{1}{2}\mathbf{W}\mathbf{s}-\frac{1}{2}\left(\mathbf{s}^T\mathbf{W}\right)^T+{\mathbf{B}_{\mathbf{u}}}^T{\boldsymbol{\xi}}^T=0, \tag{5.2.23}$$

and

$$\begin{aligned}\frac{\partial H}{\partial \boldsymbol{\xi}} &= \mathbf{B_u}\boldsymbol{\delta} + \mathbf{B_s}\mathbf{s} - \mathbf{m}_{b,r} \\ &= \mathbf{B_u}\mathbf{W}^{-1}\mathbf{W}\boldsymbol{\delta} + \mathbf{B_s}\mathbf{s} - \mathbf{m}_{b,r} = 0.\end{aligned} \tag{5.2.24}$$

Solving Equation 5.2.23 for $\mathbf{W}\boldsymbol{\delta}$ and substituting the result into Equation 5.2.24 gives

$$\mathbf{B_u}\mathbf{W}^{-1}\left(\mathbf{W}\mathbf{s} - \mathbf{B_u}^T\boldsymbol{\xi}\right) = \mathbf{m}_{b,r} - \mathbf{B_s}\mathbf{s}. \tag{5.2.25}$$

Noting that $\mathbf{s}$ lies in the null space of $\mathbf{B_u}$, Equation 5.2.25 is solved for $\boldsymbol{\xi}$ as

$$\boldsymbol{\xi} = -\left(\mathbf{B_u}\mathbf{W}^{-1}\mathbf{B_u}^T\right)^{-1}\left(\mathbf{m}_{b,r} - \mathbf{B_s}\mathbf{s}\right). \tag{5.2.26}$$

Substituting Equation 5.2.26 into Equation 5.2.23 gives

$$\mathbf{W}\boldsymbol{\delta} = \mathbf{W}\mathbf{s} + \mathbf{B_u}^T\left(\mathbf{B_u}\mathbf{W}^{-1}\mathbf{B_u}^T\right)^{-1}\left(\mathbf{m}_{b,r} - \mathbf{B_s}\mathbf{s}\right), \tag{5.2.27}$$

and simplifying gives the control input as

$$\boldsymbol{\delta} = \mathbf{s} + \mathbf{W}^{-1}\mathbf{B_u}^T\left(\mathbf{B_u}\mathbf{W}^{-1}\mathbf{B_u}^T\right)^{-1}\left(\mathbf{m}_{b,r} - \mathbf{B_s}\mathbf{s}\right) \tag{5.2.28}$$

$$= \mathbf{s} + \mathbf{B_\delta}^{\#}\left(\mathbf{m}_{b,r} - \mathbf{B_s}\mathbf{s}\right) \tag{5.2.29}$$

Observing Equation 5.2.28, we note that when none of the control effectors are saturated, $\mathbf{B_u}$ is equal to $\mathbf{B_\delta}$, and the control input is simply a weighted pseudo inverse of Equation 5.2.2 denoted by $\mathbf{B_\delta}^{\#}$. Since $\mathbf{B_\delta}$ is large on the order of 10^4 due to the dynamic pressure term, the inverse of $\mathbf{B_\delta}\mathbf{B_\delta}^T$ will be very small and potentially pose numerical issues. Therefore, the weighting matrix is selected as

$$\mathbf{W} = \left(10^4\right)\mathbf{1}_{6\times 6}. \tag{5.2.30}$$

The control effector deflection is determined iteratively at each time step by first computing the control surface deflection required to achieve the reference body moment assuming that no surfaces are saturated. If any of the control surfaces saturate, the deflection of that surface is placed into $\mathbf{s}$ and the $\mathbf{B_s}$ and $\mathbf{B_u}$ matrices are constructed. Equation 5.2.28 is then iteratively evaluated until the reference moment is achieved or all of the control surfaces saturate.

5.2.2 Angular Velocity Controller

A fast, inner loop controller tracks an angular velocity reference, $\boldsymbol{\omega}_r$, provided by the attitude controller by commanding $\mathbf{m}_{b,r}$ to the control allocation algorithm. In order to avoid feedback of the full state as required by the LQS design shown in Figure 5.1, the angular velocity controller is designed about a substate of the full state. The angular velocity states, $\mathbf{x}_f$ are obtained from the full booster state by the angular velocity output matrix, $\mathbf{C}_f$, as

$$\mathbf{x}_f = \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}} = \mathbf{C}_f \mathbf{x} = \begin{bmatrix} p & q & r \end{bmatrix}^T. \tag{5.2.31}$$

The linear state space model of the angular velocity dynamics is

$$\dot{\mathbf{x}}_f = \mathbf{A}_f \mathbf{x}_f + \mathbf{B}_f \mathbf{m}_b. \tag{5.2.32}$$

The state matrix for the angular velocity dynamics is defined as

$$\mathbf{A}_f = \frac{\partial \dot{\mathbf{x}}_f}{\partial \mathbf{x}_f} = \frac{\partial \left(\mathbf{C}_f \dot{\mathbf{x}} \right)}{\partial \left(\mathbf{C}_f \mathbf{x} \right)} = \frac{\partial \left(\mathbf{C}_f \left(\mathbf{A}\mathbf{x} + \mathbf{B}\boldsymbol{\delta} \right) \right)}{\partial \mathbf{x}_f}. \tag{5.2.33}$$

Recognising that $\mathbf{C}_f$ has full row rank, the pseudo inverse is

$$\mathbf{C}_f^{\#} = \mathbf{C}_f \left(\mathbf{C}_f \mathbf{C}_f^T \right)^{-1}. \tag{5.2.34}$$

Since the angular velocity state is simple a subset of the full state,

$$\mathbf{C}_f \mathbf{C}_f^T = \mathbf{1}_{3\times 3}, \tag{5.2.35}$$

and the pseudo inverse of the output matrix is equal to the transpose. Thus, the linear least-squares solution for $\mathbf{x}$ in Equation 5.2.31 is

$$\mathbf{x} = \mathbf{C}_f^T \mathbf{x}_f + \left(\mathbf{1} - \mathbf{C}_f^T \mathbf{C}_f \right) \mathbf{w}, \tag{5.2.36}$$

where $\mathbf{w}$ is an arbitrary vector. Substituting Equation 5.2.36 into Equation 5.2.33 gives

$$\mathbf{A}_f = \frac{\partial \left(\mathbf{C}_f \left(\mathbf{A} \left(\mathbf{C}_f^T \mathbf{x}_f + \left(\mathbf{1} - \mathbf{C}_f^T \mathbf{C}_f \right) \mathbf{w} \right) + \mathbf{B}\boldsymbol{\delta} \right) \right)}{\partial \mathbf{x}_f} = \mathbf{C}_f \mathbf{A} \mathbf{C}_f^T. \tag{5.2.37}$$

Since the control input is the body moments, the input matrix is

$$\mathbf{B}_f = \frac{\partial \dot{\boldsymbol{\omega}}_{\mathbf{B}/\mathbf{I}}}{\partial \mathbf{m}_b}. \tag{5.2.38}$$

From Equation 3.5.4 it is obvious that

$$\mathbf{B}_f = \mathbf{I}_b^{-1}. \tag{5.2.39}$$

Thus, the state equation for the angular velocity controller is

$$\dot{\mathbf{x}}_f = \mathbf{A}_f \mathbf{x}_f + \mathbf{B}_f \mathbf{m}_b = \mathbf{C}_f \mathbf{A} \mathbf{C}_f^T \mathbf{x}_f + \mathbf{I}_b^{-1} \mathbf{m}_b. \tag{5.2.40}$$

The dynamics of the fast state are then transformed into the discrete domain and the LQR cost function is selected

$$J = \frac{1}{2} \sum_{k=0}^{\infty} \left(\mathbf{x}_f^T \mathbf{Q}_f \mathbf{x}_f + \mathbf{m}_b^T \mathbf{R}_f \mathbf{m}_b \right), \tag{5.2.41}$$

where $\mathbf{Q}_f$ and $\mathbf{R}_f$ are diagonal weighting matrices. The feedback gain $\mathbf{K}_f$ is then computed as described in Section 5.1, resulting in the control law

$$\mathbf{m}_b = -\mathbf{K}_f \left(\mathbf{x}_f - \mathbf{r}_f \right) = -\mathbf{K}_f \left(\mathbf{C}_f \mathbf{x} - \mathbf{r}_f \right). \tag{5.2.42}$$

The closed loop dynamics of the full system with angular velocity feedback are

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\, \mathbf{B}_{\boldsymbol{\delta}}{}^{\#} \left[-\mathbf{K}_f \left(\mathbf{C}_f \mathbf{x} - \mathbf{r}_f \right) \right] \\ &= \left(\mathbf{A} - \mathbf{B}\, \mathbf{B}_{\boldsymbol{\delta}}{}^{\#} \mathbf{K}_f \mathbf{C}_f \right) \mathbf{x} + \mathbf{B}\, \mathbf{B}_{\boldsymbol{\delta}}{}^{\#} \mathbf{K}_f \mathbf{r}_f \\ &= \mathbf{A}_{f,cl} \mathbf{x} + \mathbf{B}_{f,cl} \mathbf{r}_f. \end{aligned} \tag{5.2.43}$$

5.2.3 Attitude Controller

The attitude controller provides angular velocity commands in order to track a slow attitude state $\mathbf{x}_s$. We would like to control motion about all three axes, and thus require three attitude states for feedback. The sideslip angle is chosen for feedback around the $\bar{z}_{\mathcal{B}}$-axis to prevent the wing from de-yawing in the oblique wing configuration due to the positive static yaw moment coefficient as shown in Figure 3.11. The roll angle is chosen for the attitude feedback about the $\bar{x}_{\mathcal{B}}$-axis to reject influence of the static roll moment coefficients, as shown in Figure 3.9, to maintain level flight. For the attitude state about the $\bar{y}_{\mathcal{B}}$-axis, both the angle of attack and the pitch angle are considered and will be selected based on the current flight control objectives. Thus, two separate attitude states are considered:

$$\mathbf{x_s} = \mathbf{C}_s \mathbf{x} = \begin{bmatrix} \Phi & \Theta & \beta \end{bmatrix}^T \tag{5.2.44}$$

and

$$\mathbf{x_s} = \mathbf{C}_s\mathbf{x} = \begin{bmatrix} \phi & \alpha & \beta \end{bmatrix}^T. \tag{5.2.45}$$

Following the same procedure as described in Equations 5.2.33-5.2.37, the state coefficient matrix is extracted from the full state as

$$\mathbf{A}_s = \frac{\partial \dot{\mathbf{x}}_s}{\partial \mathbf{x}_s} = \mathbf{C}_s\mathbf{A}\mathbf{C}_s^T. \tag{5.2.46}$$

We use a reference angular velocity as the input to the attitude states, and assume that the angular velocity controller achieves perfect tracking so that the angular velocity state may be considered as the control input. Thus, the input coefficient matrix is defined as

$$\mathbf{B}_s = \frac{\partial \dot{\mathbf{x}}_s}{\partial \mathbf{x}_f} = \frac{\partial\, \mathbf{C}_s\dot{\mathbf{x}}}{\partial \mathbf{x}_f} = \frac{\partial\, \mathbf{C}_s(\mathbf{A}\mathbf{x} + \mathbf{B}\boldsymbol{\delta})}{\partial \mathbf{x}_f}. \tag{5.2.47}$$

Substituting Equation 5.2.36 into Equation 5.2.47 gives

$$\mathbf{B}_s = \frac{\partial \left(\mathbf{C}_s\left(\mathbf{A}\left(\mathbf{C}_f^T\mathbf{x}_f + \left(\mathbf{1} - \mathbf{C}_f^T\mathbf{C}_f\right)\mathbf{w}\right)\right) + \mathbf{B}\boldsymbol{\delta}\right)}{\partial \mathbf{x}_f} = \mathbf{C}_s\mathbf{A}\mathbf{C}_f^T. \tag{5.2.48}$$

Thus, the dynamics of the attitude states are

$$\dot{\mathbf{x}}_s = \mathbf{A}_s\mathbf{x}_s + \mathbf{B}_s\mathbf{r}_f = \mathbf{C}_s\mathbf{A}\mathbf{C}_s^T\mathbf{x}_s + \mathbf{C}_s\mathbf{A}\mathbf{C}_f^T\mathbf{r}_f. \tag{5.2.49}$$

Furthermore, we would like to track the attitude reference $\mathbf{r}_s$ with zero steady state error, so we augment Equation 5.2.49 with integrator states $\mathbf{x}_{si}$. The integrator dynamics are

$$\dot{\mathbf{x}}_{si} = \mathbf{x}_s - \mathbf{r}_s, \tag{5.2.50}$$

so the augmented system becomes

$$\begin{bmatrix} \dot{\mathbf{x}}_s \\ \dot{\mathbf{x}}_{si} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_s\mathbf{A}\mathbf{C}_s^T & \mathbf{0}_{3\times3} \\ \mathbf{1}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \begin{bmatrix} \mathbf{x}_s \\ \mathbf{x}_{si} \end{bmatrix} + \begin{bmatrix} \mathbf{C}_s\mathbf{A}\mathbf{C}_f^T \\ \mathbf{0}_{3\times3} \end{bmatrix} \mathbf{r}_f + \begin{bmatrix} \mathbf{0}_{3\times3} \\ -\mathbf{1}_{3\times3} \end{bmatrix} \mathbf{r}_s. \tag{5.2.51}$$

The augmented dynamics are then transformed into the discrete domain and the feedback gain $\mathbf{K}_s = \begin{bmatrix} \mathbf{K}_{sp} & \mathbf{K}_{si} \end{bmatrix}$ is selected using LQR with the cost function

$$J = \frac{1}{2}\sum_{k=0}^{\infty}\left(\begin{bmatrix} \mathbf{x}_s^T & \mathbf{x}_{si}^T \end{bmatrix} \begin{bmatrix} \mathbf{Q}_{sp} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{Q}_{si} \end{bmatrix} \begin{bmatrix} \mathbf{x}_s \\ \mathbf{x}_{si} \end{bmatrix} + \mathbf{r}_f^T\mathbf{R}_s\mathbf{r}_f\right), \tag{5.2.52}$$

where $\mathbf{Q}_s$ and $\mathbf{Q}_{si}$ are diagonal weighting matrices for the attitude states and integrator states, respectively, and $\mathbf{R}_s$ is a diagonal weighting matrix for the angular velocity reference. The feedback control law

is then

$$\mathbf{r}_f = -\begin{bmatrix}\mathbf{K}_{sp} & \mathbf{K}_{si}\end{bmatrix}\begin{bmatrix}\mathbf{x}_s - \mathbf{r}_s \\ \mathbf{x}_{si}\end{bmatrix} = -\begin{bmatrix}\mathbf{K}_{sp} & \mathbf{K}_{si}\end{bmatrix}\begin{bmatrix}\mathbf{C}_s\mathbf{x} - \mathbf{r}_s \\ \mathbf{x}_{si}\end{bmatrix}. \tag{5.2.53}$$

The closed loop dynamics of the full state with attitude feedback are

$$\begin{bmatrix}\dot{\mathbf{x}} \\ \dot{\mathbf{x}}_{si}\end{bmatrix} = \begin{bmatrix}\mathbf{A}_{f,cl} - \mathbf{B}_{f,cl}\mathbf{K}_{sp}\mathbf{C}_s & -\mathbf{B}_{f,cl}\mathbf{K}_{si} \\ \mathbf{C}_s & \mathbf{0}_{3\times 3}\end{bmatrix}\begin{bmatrix}\mathbf{x} \\ \mathbf{x}_{si}\end{bmatrix} + \begin{bmatrix}\mathbf{B}_{f,cl}\mathbf{K}_{sp} \\ -\mathbf{I}_{3\times 3}\end{bmatrix}\mathbf{r}_s. \tag{5.2.54}$$

5.3 Outer Loop Controllers

With the attitude states controlled, outer loop single-input, single-output controllers may be designed in order to meet the primary control objectives. The primary control objectives vary with the booster configuration. In the rocket configuration, the booster must maintain zero roll angle while slowing down in order to reduce the airspeed below the critical Mach number for $\Lambda = 60°$. In the oblique wing configuration, the booster must pull up to further reduce the airspeed by exchanging kinetic energy for gravitational potential energy. In the aircraft configuration, the booster must attain the minimum sink rate glide and maintain it's heading angle. These outer loop controllers must provide a reference roll angle, pitch angle or angle of attack, and a sideslip angle to the inner loop controllers. The sideslip angle reference is zero for all cases to keep the booster aligned with the airflow in order to reduce drag and prevent the wing from de-yawing in the oblique wing configuration.

5.3.1 Rocket Configuration Controllers

Re-entry vehicles typically track a drag acceleration profile to slow down by using the angle of attack and roll angle as inputs. The angle of attack is typically set at a constant value that achieves the nominal drag acceleration profile, and the roll angle is typically the primary control input to reject disturbances. This results from the fact that rapid variations in angle of attack results in significant thermal fluctuations at high supersonic and hypersonic speeds, and therefore increases the requirements on the thermal protection system [51]. Because this work only considers the subsonic portion of the return flight, a guidance system has not been developed for the ALV to provide the drag acceleration profile. Thus, the outer loop controller in the rocket configuration simply maintains a constant angle of attack that causes the booster to generate a negative total acceleration in the $\bar{z}_{\mathcal{B}}$ direction. This requires that the angle of attack reference be greater than the equilibrium angle of attack. A constant angle of attack reference of $\alpha = 8.5°$ is selected to remain within the control surface limits throughout the subsonic descent in the rocket configuration.

5.3.2 Pull-up Manoeuvrer

Once the wing initially deploys to $\Lambda = 60°$, the booster must pull up to exchange kinetic energy for potential energy while remaining within the booster's structural limitations. Two methods are considered to achieve this pull up manoeuvre: activating the airspeed control, described in the next subsection, that commands a pitch angle reference to the attitude controller, and designing a separate flight path angle controller that commands an angle of attack reference to the attitude controller. Recalling from Equation 4.2.8, the load factor constraint is satisfied if

$$\dot{\gamma}(k) \leq \frac{g}{V_\infty(k)}(n_{max} - 1). \tag{5.3.1}$$

In level flight with a zero roll angle and a zero sideslip angle, the flight path angle is simply the difference between the pitch angle and angle of attack:

$$\gamma(k) = \Theta(k) - \alpha(k). \tag{5.3.2}$$

Pitch Angle Reference

The first method considered to perform the pull up manoeuvre is to activate the airspeed control as soon as the wing deploys to $\Lambda = 60°$. Since the booster will have a much higher airspeed than the minimum sink rate airspeed at this point, the airspeed controller will command a large pitch angle in order to slow down, thus forcing the booster to pull up. It is then assumed that the angle of attack does not vary significantly during this manoeuvre. Thus,

$$\dot{\gamma}(k) = \dot{\Theta}(k) - \dot{\alpha}(k) \approx \dot{\Theta}(k), \tag{5.3.3}$$

and the load factor limits may be respected by limiting the rate of change of the pitch angle command provided by the airspeed controller as

$$\dot{\Theta}_{max}(k) = \frac{g}{V_\infty(k)}(n_{max} - 1). \tag{5.3.4}$$

The pitch angle reference is then

$$\Theta_r(k) = \begin{cases} \Theta_{r,V}(k), & \text{if } \frac{\Theta_{r,V}(k) - \Theta(k-1)}{T_s} \leq \dot{\Theta}_{max}(k) \\ \Theta(k-1) + \dot{\Theta}_{max}(k) T_s, & \text{if } \frac{\Theta_{r,V}(k) - \Theta(k-1)}{T_s} > \dot{\Theta}_{max}(k) \end{cases} \tag{5.3.5}$$

where $\Theta_{r,V}(k)$ is the pitch angle reference provided by the airspeed controller. However, the angle of attack does vary slightly during the pull-up manoeuvre. As the velocity decrease throughout the pull-up manoeuvre, the acceptable $\dot{\gamma}$ increases and thus the angle of attack will slightly increase to pull-up faster.

Angle of Attack Reference

Another option is to design a separate outer loop controller that tracks a flight path angle reference using the angle of attack as the control input. A proportional-integral (PI) controller is designed to track a flight path angle reference γ_r as

$$\alpha_r(k) = -K_{\gamma_p}\left(\gamma(k) - \gamma_r(k)\right) - K_{\gamma_i}\sum_{n=0}^{k}\left(\gamma(n) - \gamma_r(n)\right)T_s. \tag{5.3.6}$$

The reference flight path angle may then be rate limited as

$$\dot{\gamma}_{r,max}(k) = \frac{g}{V_\infty(k)}(n_{max} - 1), \tag{5.3.7}$$

in order to limit the maximum load factor. The reference flight path angle is constructed as

$$\gamma_r(k) = \gamma(k-1) + \dot{\gamma}_{r,max}(k)T_s. \tag{5.3.8}$$

However, simulations showed that the angle of attack does vary slowly enough compared to the pitch angle such that the approximation made in Equation 5.3.3 is reasonable to make in order to remain within the booster's load factor limits. Since activating the airspeed controller after first partially deploying the wing is simpler than utilising an additional control loop, rate limiting the pitch angle during the pull-up manoeuvre is the preferred method.

5.3.3 Airspeed Controller

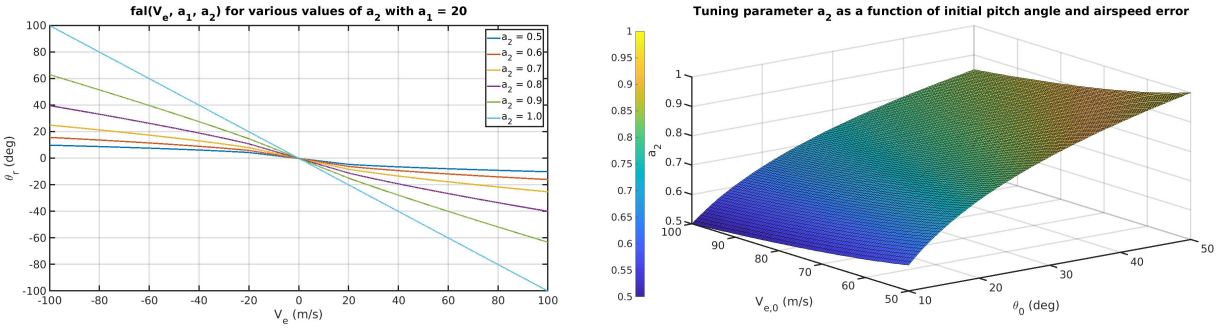
After performing the pull-up manoeuvre, the booster must attain the airspeed $V_{\infty,r}$ required for the minimum sink rate glide. In the aircraft configuration, the phugoid mode is an oscillatory mode of motion between the pitch angle and the airspeed. Therefore, the pitch angle represents a useful state to track $V_{\infty,r}$, and a SISO controller may be designed for the airspeed controller. However, the airspeed error will be very large when the airspeed controller is activated, and the booster will be at a very large flight path angle, and therefore pitch angle, immediately after the pull up manoeuvre. If a typical PID controller is implemented, the proportional term must be carefully selected to ensure that the commanded pitch angle is near the booster's pitch angle when the controller is activated to avoid large pitch-up or pitch-down commands. Selecting the proportional term to achieve this inhibits designing the controller to provide a good response when the airspeed error is small. Thus, we desire a feedback controller that saturates to some value when the error is large, namely the booster's pitch angle when the controller is activated, but still allows for independent tuning of the proportional gain when the error is small. Furthermore, it is desirable to prevent integrator wind-up at the large error values, while still achieving integral control at small error values.

Proportional Term

One attractive method, proposed by Han [52], that results in a proportional term that meets some of these requirements defines a nonlinear feedback controller function *fal* as

$$fal(e,a_1,a_2)=\begin{cases}\frac{e}{a_1^{1-a_2}}, & \text{if} |e|\leq a_1\\ |e|^{a_2}\text{sign}(e), & \text{if} |e|>a_1,\end{cases} \tag{5.3.9}$$

where a_1 and a_2 are tuning factors used to select the proportional gain, equal to $a_1^{a_2-1}$, at small errors ($|e|\leq a_1$). When the error term is large ($|e|>a_1$), setting $a_2=1$ results in a typical proportional linear controller, and choosing $a_2<1$ results in a nonlinear controller that commands a smaller control input. The *fal* function was designed to reduce overshoot of typical PID controllers to large reference step inputs, and is shown in Figure 5.5a for $a_1=20$ and various values of a_2 ranging from 0.5 to 1.0.



(a) (b)

Figure 5.5: (a) Nonlinear feedback function proposed by Han [52] to reduce overshoot to reference step inputs. (b) Surface plot showing a_2 tuning parameter required to achieve a reference Θ_0 at an airspeed error of $V_{e,0}$.

In order to utilise this function in the airspeed controller as an improvement over a linear proportional term, we would like the pitch angle reference Θ_r to be equal to the booster's pitch angle Θ_0 when the airspeed controller is activated with an error $V_{e,0}$, and for $V_{e,0}$ to be greater than a_1. This requires that

$$\Theta_r=\Theta_0=-fal(V_{e,0},a_1,a_2)=-|V_{\infty,r}-V_\infty|^{a_2}\text{sign}\left(V_{\infty,r}-V_\infty\right)=|V_{\infty,r}-V_\infty|^{a_2}. \tag{5.3.10}$$

Solving Equation 5.3.10 for a_2 gives

$$a_2=\frac{\ln(\Theta_0)}{\ln\left(|V_{\infty,r}-V_\infty|\right)}. \tag{5.3.11}$$

Figure 5.5b shows the result of Equation 5.3.11 for initial airspeed errors between $50m/s$ and $100m/s$, and initial pitch angles between 10° and 50°. As the pitch angle increases for all airspeed errors, a_2 becomes larger and therefore the *fal* function becomes more linear. This increased linearity reduces the desirable property of tuning the proportional gain at small error values while limiting the control input at large error

values. In order to preserve this ability, we propose an alternate proportional term using the hyperbolic tangent function as

$$\Theta_{r_p}(k) = -c_1 \tanh \frac{V_\infty(k) - V_{\infty,r}(k)}{c_2} \tag{5.3.12}$$

where c_1 and c_2 are tuning parameters used to set the equivalent proportional. Figure 5.6 shows this function.

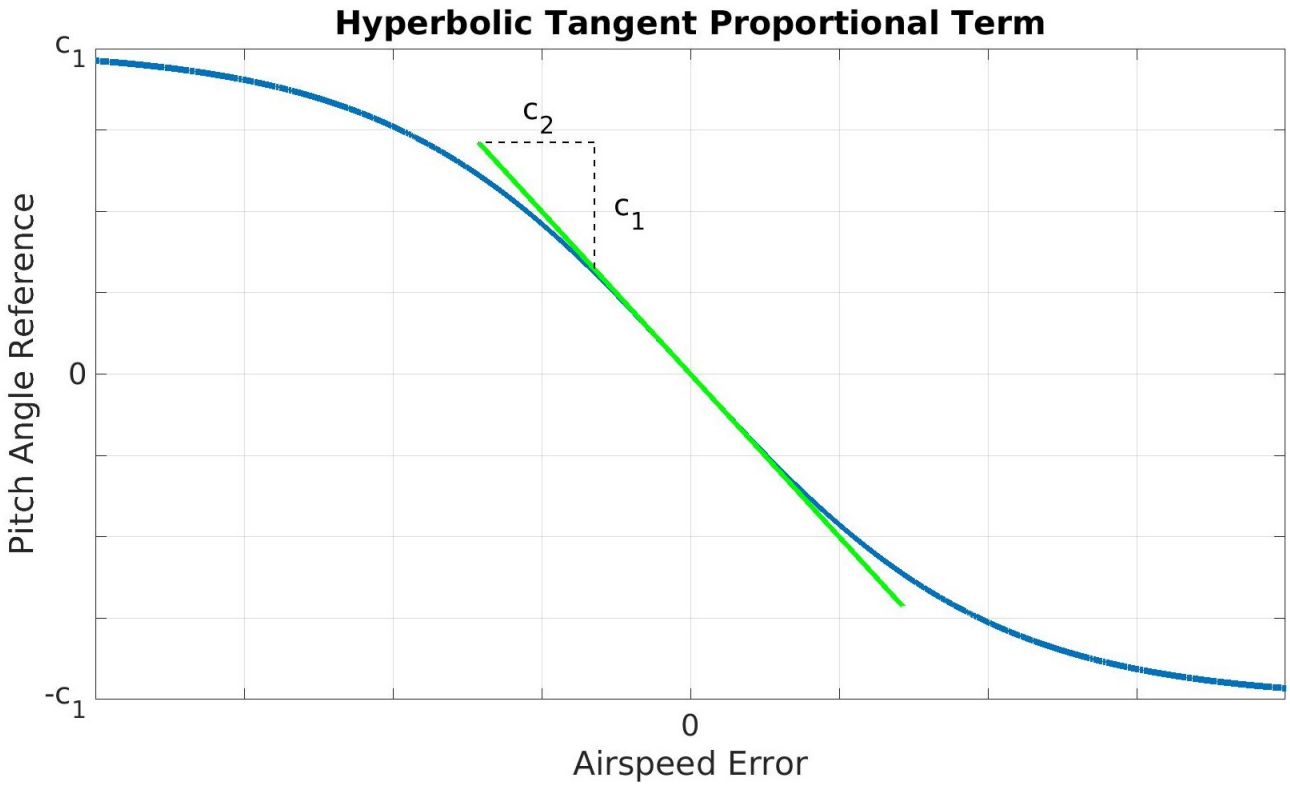


Figure 5.6: Hyperbolic tangent function used for the proportional gain in the airspeed control law .

Therefore, c_1 may be set equal to Θ_0 to saturate the control input at large error values, and the proportional term becomes linear at small error values with a gain set by

$$K_{V_p} = \frac{c_1}{c_2}. \tag{5.3.13}$$

Since the hyperbolic tangent function only approaches c_2 as the error term becomes large, there is still a trade-off between setting the gain at small error values and saturating the control input at large error values. However simulation results compare the hyperbolic tangent function and *fal* function in Section 7.2.4, and shows that the hyperbolic tangent function is able to achieve the minimum sink rate glide faster and at a higher altitude.

Integrator Term

With the proportional term limited, the problem of integrator wind up at large error values still remains. In order to handle this,

$$\Theta_{r_i} = \begin{cases} 0, & \text{if} \quad |V_\infty - V_{\infty,r}| > c_3 \\ -K_{V_i} \sum_{n=0}^{k} \left(V_\infty(n) - V_{\infty,r}(n) \right) T_s, & \text{if} \quad |V_\infty - V_{\infty,r}| <= c_3. \end{cases} \tag{5.3.14}$$

Thus, the airspeed controller is given as

$$\Theta_r = \Theta_{r_p} + \Theta_{r_i}. \tag{5.3.15}$$

The reference airspeed is the minimum sink rate airspeed for $\Lambda = 0°$.

5.3.4 Heading Angle Controller

The dynamics from heading angle to roll angle are restated here as, and is approximated at small roll angles as

$$\dot{\chi} = \frac{g}{V_\infty} \tan\Phi \approx \frac{g}{V_\infty}\Phi. \tag{5.3.16}$$

Thus, the heading angle dynamics are type 1, so a proportional controller may obtain zero steady state error. The control law is therefore

$$\Phi_r = \begin{cases} -K_\chi \left(\chi - \chi_r \right), & \text{if} \quad |\chi - \chi_r| \leq \pi \\ -K_\chi \left(2\pi + \chi - \chi_r \right), & \text{if} \quad |\chi - \chi_r| > \pi, \end{cases} \tag{5.3.17}$$

and the conditional statement ensures that the shortest turning distance to χ_r is taken. The heading angle controller is implemented after the minimum sink rate glide is obtained, and is primarily required to prevent random motions due to winds to ensure the multiple boosters do not collide while the piston engine is started.

5.3.5 Gain Scheduling

In order to handle the significant variation in the flight envelope and booster configuration, gain scheduling is performed on the gains of the multi-loop MIMO controller, and the outer loop controllers are selected according to the varying control objectives. Table 5.1 shows the controller schedule used for the outer loop controllers. The plant varies significantly with the density and compressibility of air, as well as with the sweep angle of the wing. Therefore, the altitude, Mach number and wing sweep angle are selected as the scheduling parameters for the angular velocity and attitude controller gains. For the wing sweep angles $\Lambda \in \{0°, 15°, 30°, 45°, 60°, 90°\}$, Equation 5.2.41 and Equation 5.2.52 are solved to give $\mathbf{K}_f$, $\mathbf{K}_{sp}$ and

$\mathbf{K}_{si}$ for altitudes from sea level to 11 km in 0.5 km increments, and for airspeeds from Mach 0.1 to Mach 0.8 in Mach 0.05 increments. Each element of the gain matrices is interpolated for altitudes and Mach numbers in between the gain schedule points in order to provide a smooth feedback control signal. The gains are not interpolated between the various wing sweep angles, and the wing sweep angle listed in the "Wing Sweep" column of Table 5.1 is used for a given schedule number.

Schedule Number	Wing Sweep	Θ_r or α_r	Φ_r	Scheduling Criteria
#1	$\Lambda = 90°$	$\alpha_r = 8.5°$	$\Phi_r = 0$	$M <= 0.80$
#2	$\Lambda = 60°$	$\Theta_r = \Theta(M = 0.65)$	$\Phi_r = 0$	$M <= 0.65$
#3	$\Lambda = 60°$	$\Theta_r =$ Eq. 5.3.5	$\Phi_r = 0$	$\Lambda = 60°$
#4	$\Lambda = 45°$	$\Theta_r =$ Eq. 5.3.12	$\Phi_r = 0$	$M <= 0.45$
#5	$\Lambda = 30°$	$\Theta_r =$ Eq. 5.3.12	$\Phi_r = 0$	$M <= 0.25$
#6	$\Lambda = 15°$	$\Theta_r =$ Eq. 5.3.12	$\Phi_r = 0$	$M <= 0.19$
#7	$\Lambda = 0°$	$\Theta_r =$ Eq. 5.3.12	$\Phi_r =$ Eq. 5.3.17	$M <= 0.17$

Table 5.1: Outer loop controller schedule.

In the rocket configuration $\Lambda = 90°$, the constant angle of attack reference of 8.5° is provided to the attitude controller until the booster reaches Mach 0.65. Schedule #2 is the transition between the rocket configuration, $\Lambda = 90°$, and the oblique wing configuration, $\Lambda = 60°$. The booster attempts to maintain a constant attitude throughout the deployment, thus $\Theta(M = 0.65)$ is the booster's pitch angle at the controller scheduling switch point. Once the wing reaches $\Lambda = 60°$ (schedule #3), the airspeed controller is activated, which provides a large pitch angle reference to the attitude controller. The rate of change of this pitch angle reference is rate limited to satisfy the load factor constraints of the booster during the pull-up. Furthermore, the nonlinear hyperbolic proportional term is utilised to saturate the pitch angle reference to a maximum of 50°. The scheduling criteria for schedule #4 to #7 are selected as functions of the Mach number. Although they the booster is well below the critical Mach number of the fully deployed wing after the pull-up manoeuvre, immediately deploying the wing at this point is limited by the minimum angle of attack of −5° as restricted by the airfoil. At the end of the pull up manoeuvre, the booster has a very large flight path angle and must obtain near level flight to track the minimum sink rate airspeed. This requires that the booster maintain an angle of attack below trim in order to pitch down. Since the wing produces more lift at the lower sweep angles, fully deploying the wing at the end of the pull-up manoeuvre will cause a significant, negative angle of attack that may stall the wing. These design choices are compared in Section 7.2.

5.4 Conclusion

This chapter has presented the design of the flight control system for the subsonic wing deployment of the ALV. The significant variation in the booster's configuration is handled through the use of a cascaded control architecture. At the lowest level, a control allocation algorithm handles the variation in effec-

tiveness and availability of the control surfaces. An angular velocity controller then handles the inertial cross-coupling introduced by the oblique wing configuration by tracking an angular velocity reference by providing a moment command to the control allocation algorithm. With the angular velocity controlled, an attitude controller the commands this angular velocity in order to track an attitude reference relative to inertial space or to the velocity vector. This attitude reference is constructed from various sources: a constant zero sideslip reference is provided to keep the booster aligned with the airflow; a roll angle reference is provided to either maintain level flight or to track a heading angle reference; and a pitch angle or angle of attack reference is provided to reduce the velocity of the booster and ultimately obtain a minimum sink rate glide. This cascaded control architecture forms the primary contribution of this research.

Chapter 6

State Estimator Design

The flight controllers presented in Chapter 5 assumes that all the states are available for feedback. This chapter describes the sensors required to measure or to estimate these states. The required states are the polar velocity (airspeed, sideslip angle and angle of attack), the attitude of booster parameterised by the 3-2-1 Euler angle set, the body angular velocity, the flight path angle and the heading angle. Furthermore, the Mach number is required to initiate the deployment of the wing.

This chapter first describes the sensors considered and modelled in this work, which together constitute an inertial navigation system (INS) and an air data computer. The inertial navigation system provides measurements to directly or indirectly sense the booster's inertial position, inertial velocity, attitude and angular velocity. The air data computer determines the polar velocity and the Mach number.

6.1 Sensors

The sensors suite for the rocket booster must be low cost, therefore commercially available sensors are utilised. Microelectromechanical systems (MEMS) sensors are commonly used in low-cost applications, and thus make up the primary inertial sensors. Sensors that measure the boosters motion relative to the air are also selected.

6.1.1 Inertial Navigation System

The inertial navigation system consists of an inertial measurement unit (IMU), a magnetometer and a GPS receiver. The IMU modelled is the Analog Devices ADIS16355, which consists of a three-axis accelerometer and a three-axis gyroscope. The IMU has a $\pm 10g$ measurement range about all three axes, and a temperature range of $-40°C$ to $+85°C$ [53]. The magnetometer modelled is the Honeywell HMC1053, which has a operating temperature range of $-40°C$ to $+125°C$. The GPS receiver modelled is the u-blox TIM-4H.

Accelerometer

The accelerometer measures the total acceleration in the body frame, which is the sum of the dynamic acceleration, $\ddot{\mathbf{r}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}$ and static acceleration due to gravity $\mathbf{a}^{\mathcal{B}}_{static}$ as

$$\mathbf{a}^{\mathcal{B}}_{meas} = \ddot{\mathbf{r}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} + \mathbf{a}^{\mathcal{B}}_{static}. \tag{6.1.1}$$

The accelerometer measurements are corrupted by noise and bias drift. The noise density for the accelerometer is 1.85 $mg/\sqrt{Hz}$ rms and has a velocity random walk of 2 $m/s/\sqrt{hr}$ [53].

Gyroscope

The gyroscope directly measures the inertial angular velocity of the body frame, $\boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}$, and is also corrupted by measurement noise and bias drift. The gyroscope has a noise density of 0.05 $deg/s/\sqrt{(Hz)}$ and an angular random walk from is 4.2 $^\circ/\sqrt{hr}$ [53].

Magnetometer

The magnetometer provides a measurement of the Earth's magnetic field $\mathbf{b}^{\mathcal{B}}$ in the body frame. The magnetometer modelled has a noise density of 5×10^{-5} $gauss/\sqrt{Hz}$. Since the bias from the magnetometer is dependent on it's location within the booster, calibration will be required after installation to zero the biases.

GPS

The GPS provides unbiased measurements of the rocket booster's inertial position and inertial velocity. The GPS modelled is the uBlox TIM-4H. The Circular Error Probability (CEP) is given as 2.5 m and the Spherical Error Probability (SEP) is 5 m [54]. The CEP and SEP are converted to standard deviations in the North, East and Down directions as

$$\begin{aligned} CEP &= 0.59(\sigma_N + \sigma_E), \\ SEP &= 0.5(\sigma_N + \sigma_E + \sigma_D). \end{aligned} \tag{6.1.2}$$

The standard deviations in Equation 6.1.2 are determined by using the CEP to solve σ_N and σ_E assuming the horizontal accuracy is circular, then using the SEP to solve for σ_D. Measurements on the GPS are modelled as white noise with the determined standard deviations. The velocity measurements are modelled with random noise with a standard deviation of 0.5 m/s [25]. The GPS also contains a measurement latency due to the required processing. The latency will depend on the specific settings of the GPS receiver and must be determined on the actual hardware used, however the latency on the velocity measurement

is typically much greater than the latency on the position measurement since the velocity is calculated from the phase of the carrier signal. The delays determined in [22] are used in simulation, which is a 0.1 s delay on the position measurement and a 1.1 s delay on the velocity measurement.

6.1.2 Air Data Computer

An air data computer is required to determine the Mach number, airspeed, angle of attack and sideslip angle. Measurements of these states are typically obtained through the use of pressure ports or wind vanes.

Airspeed and Wind Angle Sensors

The airspeed, sideslip angle and angle of attack may be directly measured with wind angle sensors. Typically, aircraft utilise air probes, such as pitot-static probes or wind vanes, mounted externally to the vehicle and exposed to the airflow in order to obtain these states. However, this is problematic for the ALV as the hypersonic phase of re-entry causes extreme heating to any surface exposed to the flow. Deployable air data sensors were utilised on the Space Shuttle, and the angle of attack and sideslip angle were estimated using inertial data by assuming that atmospheric winds were minimal at the higher altitudes in which the hypersonic phase of re-entry occurs. This, however, increases mechanical complexity resulting in additional mass, cost and risk. An alternative method for sensing the wind angles is utilising a matrix of pressure ports integrated on the nose of the booster, and is known as a Flush Air Data Sensing (FADS) system [34].

FADS have been designed for many hypersonic and re-entry vehicles due to the minimal impact on the vehicle design and inherent protection from heating, such as NASA's space shuttle [34] and the X-33 [55] suborbital spaceplane. The FADS relies on obtaining an aerodynamic model of the flow over the nose of the vehicle for various flight conditions. The pressure measured at each port can then be used to determine the relative air velocity. Several methods have been proposed for determining the flow conditions from the measured pressures, such as nonlinear regression [55], neural networks [56] and geometrical pressure variation models [57]. A FADS is the recommended solution for the first stage booster, however the design and implementation of a FADS for the ALV is not within the scope of this work due to the complexity required to model the airflow and implement an algorithm. Thus, this thesis assumes that wind vanes, such as the devices from Aerosonic Corporation shown in Figure 6.1, are available to directly measure the wind angles. These sensors have a calibrated accuracy of $\pm 0.25^\circ$, which is modelled as a constant bias in the simulation results presented.

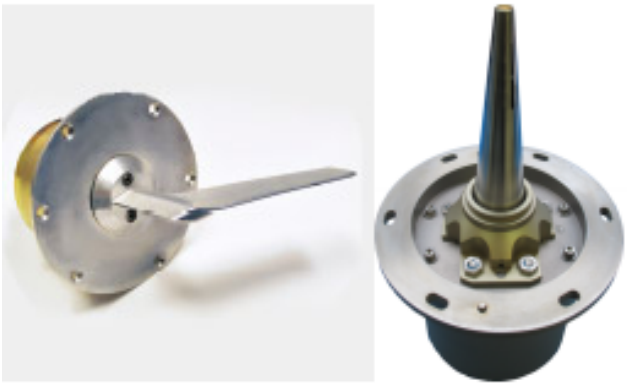


Figure 6.1: Pressure driven wind angle sensor.

Pressure Sensors

Pressure sensors are used to measure the total pressure P_T and the static pressure P_S. The difference between the two is the dynamic pressure Q_∞. The total pressure occurs at stagnation points in the airflow, and thus is typically measured at the nose of the vehicle or from a pitot-static probe. For incompressible flow ($M < 0.3$) the relation between airspeed and dynamic pressure is

$$Q_\infty = P_T - P_S = \frac{1}{2}\rho_\infty V_\infty^2, \tag{6.1.3}$$

where ρ_∞ is the free-stream air density. For compressible flow, the variation in density results in the relation

$$\frac{P_T}{P_S} = \left[1 + 0.2\frac{V_\infty^2}{a^2}\right]^{3.5}. \tag{6.1.4}$$

The speed of sound a is

$$a = \sqrt{\kappa R T_\infty}, \tag{6.1.5}$$

where T_∞ is the free-stream air temperature, κ is the ratio of specific heats assumed to be a constant value of 1.4 for air, and R is the ideal gas constant assumed to be 269.7 J/kg/K for air. The Mach number can be obtained directly from P_T and P_S through the relation for subsonic speeds

$$\frac{P_T}{P_S} = \left(1 + 0.2M^2\right)^{3.5}. \tag{6.1.6}$$

The total pressure occurs at the stagnation points in the airflow, and may be assumed to be the largest pressure measured by the FADS [57]. The static pressure is typically obtained from a pressure port on the side of the fuselage or of the pitot-static probe.

6.2 State Estimators

Estimators have been implemented in order to fuse measurements from the sensors to obtain the states. A block diagram of the estimators is shown in Figure 6.2. Many works in the literature have shown that separating the task of determining the attitude and the inertial position and velocity of an aircraft into separate estimators reduces computational time with minimal impact on the estimation accuracy [25][21][22], as describe in the literature review. This work implements the estimator developed in the ESL by Bijker [25], which was designed for an airship and used separated Kalman filters to estimate the attitude and the inertial position and velocity. A Multiplicative Extended Kalman Filter (MEKF) was used for the attitude estimator, and a separate Kalman filter was used for the inertial position and velocity estimator. A crucial assumption in this design was that the airship was not accelerating, thus allowing the accelerometer to be used as a measurement of gravity in the MEKF. The primary contribution of the state estimator designed in this work is to account for this the significant dynamic acceleration that will occur during the deployment of the wing within the MEKF framework.

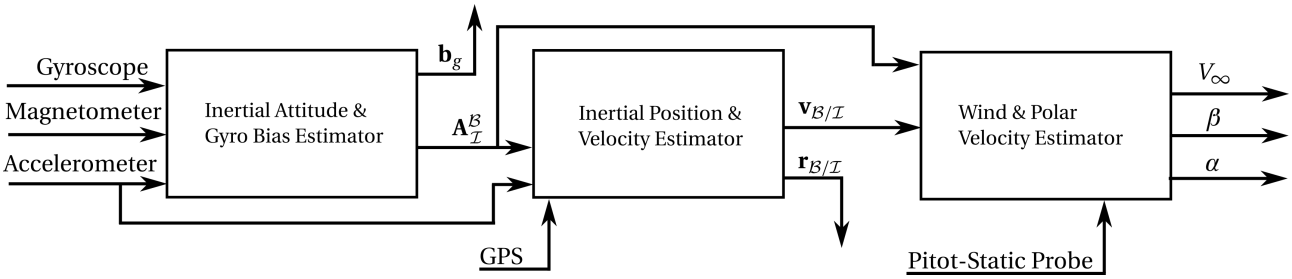


Figure 6.2: Estimator Block Diagram

6.2.1 Inertial Position and Velocity Estimator

A simple Kalman filter from [40] has been implemented to obtain the inertial position and velocity, which are estimated by propagating measurements from the accelerometer and correcting with GPS measurements. The state to be estimated is $\mathbf{x} = \begin{bmatrix} \mathbf{r}_{\mathcal{B}}^{\mathcal{I}} & \dot{\mathbf{r}}_{\mathcal{B}}^{\mathcal{I}} \end{bmatrix}^T$, with the dynamic acceleration as input. The discrete state space model is

$$\begin{bmatrix} \mathbf{r}_{\mathcal{B},k+1}^{\mathcal{I}} \\ \dot{\mathbf{r}}_{\mathcal{B},k+1}^{\mathcal{I}} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_{3\times3} & T_s\mathbf{1}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{1}_{3\times3} \end{bmatrix} \begin{bmatrix} \mathbf{r}_{\mathcal{B},k}^{\mathcal{I}} \\ \dot{\mathbf{r}}_{\mathcal{B},k}^{\mathcal{I}} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times3} \\ T_s\mathbf{1}_{3\times3} \end{bmatrix} \ddot{\mathbf{r}}_{\mathcal{B},k}^{\mathcal{I}}. \tag{6.2.1}$$

The accelerometer measures total accelerations in the body frame. Thus, the accelerometer output must be rotated into the NED frame using the current estimate of the attitude obtained from the MEKF, and the static acceleration must be removed. Assuming a constant gravity field with a gravitational acceleration

of $g = 9.81$ m/s in the down direction results in

$$\ddot{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k} = \left[\hat{\mathbf{A}}^{\mathcal{B}}_{\mathcal{I},k}\right]^{-1} \mathbf{a}^{\mathcal{B}}_{meas,k} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}. \tag{6.2.2}$$

The GPS provides a direct measurement of the state, so the filter measurement is

$$\mathbf{y}_k = \mathbf{1}_{6\times 6} \begin{bmatrix} \mathbf{r}^{\mathcal{I}}_{\mathcal{B},k} \\ \dot{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k} \end{bmatrix}. \tag{6.2.3}$$

A Kalman filter [20] with a fixed gain $\mathbf{L}$ is used to estimate the state, which is solved for using the measurement noise covariance of

$$\mathbf{R} = \mathbf{1}_{6\times 6}, \tag{6.2.4}$$

and process noise covariance of

$$\mathbf{Q} = \nu \mathbf{R}, \tag{6.2.5}$$

with the tuning parameter ν selected as $\nu = 10$ based on simulation results. The delay on the GPS measurements is significant because of the high velocity of the booster, and must be considered in the Kalman filter. One method to handle delayed measurements is to propagate the state backwards in time to the sample that corresponds with the measurement, apply the innovation and then propagate the state forward to the current sample [58]. This is computationally intensive, and would require storing the accelerometer input, kinematic state, and attitude estimate in memory for the duration of the delay. Furthermore, the attitude estimator would also need to be re-propagated when using gravity as a reference vector. Another method estimates the delayed states as part of the Kalman filter, adding a state for each delayed measurement [25]. This would double the size of the kinematic Kalman filter and is thus not ideal, although it may provide better results if the measurement delay is not constant. A simple approach that does not increase the computational intensity is to use the state corresponding to the delayed measurement in the innovation for the current time step [29]. This approach does require storing the delayed state in memory, and assumes that the delay is known and constant. Using this method, the measurement update for the Kalman filter is

$$\hat{\mathbf{x}}_{k+1} = \begin{bmatrix} \hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k+1} \\ \hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k+1} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k} \\ \hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k} \end{bmatrix} + \mathbf{L}\left(\mathbf{y}_k - \begin{bmatrix} \hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k-z_1} \\ \hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k-z_2} \end{bmatrix} \right), \tag{6.2.6}$$

where z_1 is the number of samples that the position measurement is delayed by and z_2 is the number of samples that the velocity measurement is delayed by. For the GPS with a 0.1 s position delay and a 1.1 s velocity delay, $z_1 = 5$ and $z_2 = 55$. Since the GPS has a low update rate of 4 Hz, Equation 6.2.6 is only

applied if the GPS updates within the last 20 ms time step at which the estimated is propagated at.

6.2.2 Inertial Attitude Estimator

The booster's attitude and the gyroscope bias are estimated using a quaternion based MEKF [23], where the gyroscope's angular velocity measurements propagate the attitude. In the MEKF, measurements of a known inertial vector are made in the body frame, and compared with a model of that inertial vector rotated into the body frame using the current attitude estimate. This provides an estimated perturbation from the current state estimate, which, for small perturbations, allows for only the vector part of the quaternion to be estimated. The inertial vectors considered for measurement in this work are the magnetic field vector, the gravity vector, and the booster's velocity vector. The estimator implemented here is based on the MEKF developed for an airship in [25], with a difference that the dynamic accelerations of the booster must be considered in order to utilise separate Kalman filters for the attitude and the position and velocity.

State Dynamics

The dynamics of the quaternion are given as

$$\dot{\mathbf{q}} = \frac{1}{2}\boldsymbol{\omega} \otimes \mathbf{q}, \tag{6.2.7}$$

where

$$\boldsymbol{\omega} = \begin{bmatrix} \boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}^{\mathcal{B}} \\ 0 \end{bmatrix} = \begin{bmatrix} P \\ Q \\ R \\ 0 \end{bmatrix}. \tag{6.2.8}$$

A gyroscope obtains a measurement $\mathbf{u}$ of the true angular velocity corrupted by noise, $\boldsymbol{\eta_1}$, and a bias, $\boldsymbol{\epsilon}_g$. Thus,

$$\boldsymbol{\omega}_{\mathcal{B}/\mathcal{I}}^{\mathcal{B}} = \mathbf{u} - \boldsymbol{\epsilon}_g - \boldsymbol{\eta_1}. \tag{6.2.9}$$

The gyroscope bias is modelled as a random walk process as

$$\dot{\boldsymbol{\epsilon}}_\mathbf{g} = \boldsymbol{\eta_2}. \tag{6.2.10}$$

The noises are modelled as Gaussian white noise with zero mean:

$$\boldsymbol{\eta_1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q_1}) \tag{6.2.11}$$

and

$$\boldsymbol{\eta_2} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q_2}). \tag{6.2.12}$$

The quaternion can be written in terms of the perturbation quaternion, $\boldsymbol{\delta q}$, and the quaternion estimate as

$$\mathbf{q} = \boldsymbol{\delta q} \otimes \hat{\mathbf{q}}. \tag{6.2.13}$$

Assuming that the perturbation quaternion is small, the angle element of the perturbation quaternion will not have any dynamics. Thus

$$\delta \dot{q}_4 = 0, \tag{6.2.14}$$

and the dynamics of the vector perturbation can be shown to be

$$\boldsymbol{\delta \dot{q}}_{vec} = \boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} \times \boldsymbol{\delta q}_{vec} - \frac{1}{2}(\boldsymbol{\delta \epsilon}_g + \boldsymbol{\eta_1}). \tag{6.2.15}$$

Thus, the MEKF only needs to estimate the vector part of the perturbation quaternion, and the fourth element can be calculated to satisfy the unity norm constraint as

$$\delta q_4 = \sqrt{1 - \|\boldsymbol{\delta q}_{vec}\|^2}. \tag{6.2.16}$$

The state vector is then

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} & \boldsymbol{\epsilon}_g \end{bmatrix}^T, \tag{6.2.17}$$

and the perturbation vector is

$$\boldsymbol{\delta x} = \begin{bmatrix} \boldsymbol{\delta q}_{vec} & \boldsymbol{\delta \epsilon}_g \end{bmatrix}^T. \tag{6.2.18}$$

The state space model is

$$\dot{\boldsymbol{\delta x}} = \mathbf{F}\boldsymbol{\delta x} + \mathbf{G}\mathbf{s} = \begin{bmatrix} \left[\boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} \times\right] & -\frac{1}{2}\mathbf{1}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta q}_{vec} \\ \boldsymbol{\delta \epsilon}_g \end{bmatrix} + \mathbf{1}_{6\times 6} \begin{bmatrix} -\frac{1}{2}\boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{bmatrix}, \tag{6.2.19}$$

where $\left[\boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} \times\right]$ is the skew symmetric matrix operator.

Discretization

The continuous state space model must be converted to discrete for implementation on a microcontroller. The discrete state transition matrix for the perturbation is

$$\mathbf{\Phi} = e^{\mathbf{F}t} \approx \mathbf{1}_{6\times 6} + T_s\mathbf{F} + \frac{1}{2!}T_s^2\mathbf{F}^2 + \frac{1}{3!}T_s^3\mathbf{F}^3 + \ldots \tag{6.2.20}$$

The terms of order 2 and higher may be ignored for $T_s = 0.02$ s. The continuous process noise must also be converted to discrete, which is given as

$$\mathbf{Q}_k = \int_0^{T_s} \mathbf{\Phi}(\tau)\mathbf{G}\,\mathbf{Q}_c\,\mathbf{G}^T\mathbf{\Phi}^T(\tau)d\tau. \tag{6.2.21}$$

Assuming that the attitude dynamics are significantly slower than the sampling period, the discrete process noise is given as

$$\mathbf{Q}_k \approx T_s\mathbf{\Phi}\mathbf{G}\,\mathbf{Q}_c\,\mathbf{G}^T\mathbf{\Phi}^T, \tag{6.2.22}$$

where

$$\mathbf{Q}_c = E[\mathbf{s}\mathbf{s}^T] = \begin{bmatrix} \frac{1}{4}\mathbf{Q}_1 & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{Q}_2 \end{bmatrix} \tag{6.2.23}$$

assuming that the gyro measurement noise and random walk process noise are uncorrelated.

Measurement Update

The innovation is

$$\mathbf{e} = \boldsymbol{\nu}^{\mathcal{B}}_{meas} \times \hat{\boldsymbol{\nu}}^{\mathcal{B}}_{model} = \boldsymbol{\nu}^{\mathcal{B}}_{meas} \times \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\hat{\mathbf{q}})\right]\boldsymbol{\nu}^{\mathcal{I}}_{model}. \tag{6.2.24}$$

where $\boldsymbol{\nu}_{meas}$ is measured unit vector and $\boldsymbol{\nu}_{model}$ is the modeled reference inertial unit vector. It can be shown that

$$\left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\mathbf{q})\right] = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\boldsymbol{\delta}\mathbf{q})\right]\left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\hat{\mathbf{q}})\right]. \tag{6.2.25}$$

where

$$\left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\boldsymbol{\delta}\mathbf{q})\right] \simeq \begin{bmatrix} 1 & 2\delta q_3 & -2\delta q_2 \\ -2\delta q_3 & 1 & 2\delta q_1 \\ 2\delta q_2 & -2\delta q_1 & 1 \end{bmatrix} \tag{6.2.26}$$

when the perturbation quaternion is very small. Thus, $\boldsymbol{\nu}^{\mathcal{B}}_{meas}$ can be expressed as

$$\boldsymbol{\nu}^{\mathcal{B}}_{meas} = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\mathbf{q})\right]\boldsymbol{\nu}^{\mathcal{I}}_{model} = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\boldsymbol{\delta}\mathbf{q})\right]\left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\hat{\mathbf{q}})\right]\boldsymbol{\nu}^{\mathcal{I}}_{model} = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\boldsymbol{\delta}\mathbf{q})\right]\hat{\boldsymbol{\nu}}^{\mathcal{B}}_{model}. \tag{6.2.27}$$

Using Equation 6.2.27 in Equation 6.2.24 results in

$$\mathbf{e} = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\boldsymbol{\delta}\mathbf{q})\right]\hat{\boldsymbol{\nu}}^{\mathcal{B}}_{model} \times \hat{\boldsymbol{\nu}}^{\mathcal{B}}_{model} = 2\begin{bmatrix} \delta q_1(\nu_y^2+\nu_z^2) - \delta q_2\,\nu_x\,\nu_y - \delta q_3\,\nu_x\,\nu_z \\ -\delta q_1\,\nu_x\,\nu_y + \delta q_2(\nu_x^2+\nu_z^2) - \delta q_3\,\nu_y\,\nu_z \\ -\delta q_1\,\nu_x\,\nu_z - \delta q_2\,\nu_y\,\nu_z + \delta q_3(\upsilon\,\nu_x^2+\nu_y^2) \end{bmatrix}. \tag{6.2.28}$$

This result shows that the innovation can be seen as a function of the perturbation quaternion. The measurement matrix required for the EKF algorithms is then computed as

$$\mathbf{H} = \left.\frac{\partial \mathbf{e}}{\partial \boldsymbol{\delta}\mathbf{x}}\right|_{\boldsymbol{\delta}\hat{\mathbf{x}}} = \left[\left.\frac{\partial \mathbf{e}}{\partial \boldsymbol{\delta}\mathbf{q}}\right|_{\boldsymbol{\delta}\hat{\mathbf{q}}} \quad \mathbf{0}_{3\times 3}\right]. \tag{6.2.29}$$

The partial derivative in Equation 6.2.29 is evaluated as

$$\left.\frac{\partial \mathbf{e}}{\partial \boldsymbol{\delta}\mathbf{q}}\right|_{\boldsymbol{\delta}\hat{\mathbf{q}}} = 2\begin{bmatrix} \hat{\nu}_y^2+\hat{\nu}_z^2 & -\hat{\nu}_x\,\hat{\nu}_y & -\hat{\nu}_x\,\hat{\nu}_z \\ -\hat{\nu}_x\,\hat{\nu}_y & \hat{\nu}_x^2+\hat{\nu}_z^2 & -\hat{\nu}_y\,\hat{\nu}_z \\ -\hat{\nu}_x\,\hat{\nu}_z & -\hat{\nu}_y\,\hat{\nu}_z & \hat{\nu}_x^2+\hat{\nu}_y^2 \end{bmatrix}. \tag{6.2.30}$$

MEKF Algorithm

The MEKF is operated at 50 Hz as follows:

1. Propagate the state estimate:

$$\hat{\mathbf{q}}_{k+1|k} = \frac{T_s}{2}\hat{\boldsymbol{\omega}}_k \otimes \hat{\mathbf{q}}_{k|k} + \hat{\mathbf{q}}_{k|k}$$

$$\hat{\boldsymbol{\omega}}_k = \mathbf{u}_k - \hat{\boldsymbol{\epsilon}}_{k|k}$$

2. Propagate the perturbation state estimate covariance:

$$\mathbf{P}_{k+1|k} = \boldsymbol{\Phi}_{k+1|k}\,\mathbf{P}_{k|k}\boldsymbol{\Phi}^T_{k+1|k} + \mathbf{Q}_{k+1}$$

3. Compute the predicted model vector in body coordinates:

$$\hat{\boldsymbol{\nu}}^{\mathcal{B}}_{model} = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}(\hat{\mathbf{q}}_{k+1|k})\right]\boldsymbol{\nu}^{\mathcal{I}}_{model}$$

in order to obtain the innovation $\mathbf{e}_{k+1}$ using Equation 6.2.24 and the measurement matrix $\mathbf{H}_{k+1|k}$ using Equation 6.2.30 in Equation 6.2.29.

4. Compute the gain:

$$\mathbf{L}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{H}_{k+1|k}^{T}\left(\mathbf{H}_{k+1|k}\mathbf{P}_{k+1|k}\mathbf{H}_{k+1|k}^{T} + \mathbf{R}\right)^{-1} = \begin{bmatrix} \mathbf{L}_{\boldsymbol{\delta}\mathbf{q},k+1} \\ \mathbf{L}_{\boldsymbol{\delta}\mathbf{b},k+1} \end{bmatrix}$$

5. Update the perturbation state estimates:

$$\boldsymbol{\delta}\hat{\mathbf{q}}_{k+1} = \mathbf{L}_{\boldsymbol{\delta}\mathbf{q},k+1}\mathbf{e}_{k+1}$$

$$\boldsymbol{\delta}\hat{\mathbf{b}}_{k+1} = \mathbf{L}_{\boldsymbol{\delta}\mathbf{b},k+1}\mathbf{e}_{k+1}$$

and apply the quaternion norm constraint:

$$\boldsymbol{\delta}\hat{\mathbf{q}}_{k+1} = \begin{bmatrix} \boldsymbol{\delta}\hat{\mathbf{q}}'_{k+1} \\ \sqrt{1 - \|\boldsymbol{\delta}\hat{\mathbf{q}}'_{k+1}\|^2} \end{bmatrix}$$

6. Update the state estimates:

$$\hat{\mathbf{q}}_{k+1|k+1} = \boldsymbol{\delta}\hat{\mathbf{q}}_{k+1} \otimes \hat{\mathbf{q}}_{k+1|k}$$

$$\hat{\boldsymbol{\epsilon}}_{k+1|k+1} = \hat{\boldsymbol{\epsilon}}_{k|k} + \boldsymbol{\delta}\hat{\boldsymbol{\epsilon}}_{k+1}$$

7. Compute the estimated model vector in body coordinates:

$$\hat{\boldsymbol{\nu}}_{model,k+1}^{\mathcal{B}} = \left[\mathbf{A}_{\mathcal{I}}^{\mathcal{B}}(\hat{\mathbf{q}}_{k+1|k+1})\right]\boldsymbol{\nu}_{model,k}^{\mathcal{I}}$$

in order to obtain $\mathbf{H}_{k+1|k+1}$. Update the state estimate covariance using the Joseph form for numerical stability:

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{1}_{6\times6} - \mathbf{L}_{k+1}\mathbf{H}_{k+1|k+1}\right)\mathbf{P}_{k+1|k}\left(\mathbf{1}_{6\times6} - \mathbf{L}_{k+1}\mathbf{H}_{k+1|k+1}\right)^{T} + \mathbf{L}_{k+1}\mathbf{R}\,\mathbf{L}_{k+1}^{T}$$

Steps 3 to 7 above are repeated for each measurement available, with subsequent measurements using $\hat{\mathbf{x}}_{k+1|k+1}$ and $\mathbf{P}_{k+1|k+1}$ from the previous measurement in place of $\hat{\mathbf{x}}_{k+1|k}$ and $\mathbf{P}_{k+1|k}$. Three inertial vectors are considered for use in the MEKF: the magnetic field unit vector $\boldsymbol{\nu}_{mag}^{\mathcal{I}}$, the gravity field unit vector $\boldsymbol{\nu}_{grav}^{\mathcal{I}}$, and the velocity unit vector $\boldsymbol{\nu}_{vel}^{\mathcal{I}}$.

Magnetometer Measurement

The inertial magnetic field vector $\mathbf{b}^{\mathcal{I}}$ is obtained from the 12th generation of the Internation Geomagnetic Reference Field (IGRF12) model [59]. The IGRF12 provides the predicted magnetic field as a function of latitude, longitude and distance from the Earth's core. The model uses 2015.0 as the epoch, and predicts

secular variations in the magnetic field until 2020.0. The reference vector for the MEKF is

$$\boldsymbol{\nu}^{\mathcal{I}}_{mag,k} = \frac{\mathbf{b}^{\mathcal{I}}_{k}}{\|\mathbf{b}^{\mathcal{I}}_{k}\|}. \tag{6.2.31}$$

The magnetic field vector **b** can be directly measured in the body frame with a magnetometer to provide the measurement vector as

$$\boldsymbol{\nu}^{\mathcal{B}}_{mag} = \frac{\mathbf{b}^{\mathcal{B}}_{meas,k}}{\|\mathbf{b}^{\mathcal{B}}_{meas,k}\|}. \tag{6.2.32}$$

A Simulink block implementing the IGRF12 is available in the Simulink Aerospace Blockset, and is used in the simulation.

Gravity Field Measurement

For the flyback portion of flight, the inertial gravity vector will be constantly in the downward direction with a near constant magnitude of 9.81 m/s. Thus, the reference vector for the MEKF is assumed to be

$$\boldsymbol{\nu}^{\mathcal{I}}_{grav} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{6.2.33}$$

In order to obtain an measurement of the gravity vector in the body frame, the dynamic acceleration must be subtracted from the accelerometer measurement. From Equation 6.1.1, the gravity vector is the static acceleration given as

$$\mathbf{g}^{\mathcal{B}}_{k} = \mathbf{a}^{\mathcal{B}}_{meas,k} - \ddot{\mathbf{r}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I},k}. \tag{6.2.34}$$

The measurement unit vector used in the MEKF is then

$$\boldsymbol{\nu}^{\mathcal{B}}_{grav,k} = \frac{\mathbf{g}^{\mathcal{B}}_{k}}{\|\mathbf{g}^{\mathcal{B}}_{k}\|}. \tag{6.2.35}$$

Thus, the dynamic acceleration $\ddot{\mathbf{r}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}$ must be determined from the available sensor measurements or state estimates. One method considered is to differentiate $\dot{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I}}$ from the inertial position and velocity estimator and rotate the result into the body frame. The dynamic acceleration term in Equation 6.2.34 is then

$$\ddot{\mathbf{r}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I},k} = \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}\left(\hat{\mathbf{q}}_{k|k-1}\right)\right]\frac{\hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I},k} - \hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I},k-1}}{T_s}. \tag{6.2.36}$$

Note that this requires differentiating the output of the inertial position and velocity Kalman filter, where the gain has been selected to heavily trust the GPS measurement since it is unbiased. This causes a discrete jump in the velocity output every 250 ms, resulting in a very noisy measurement of the dynamic acceler-

ation due to differentiating the filter output. Two options have been considered to reduce this noise; the dynamic acceleration may be low pass filtered, or the gravity field measurement can simply be skipped every 250 ms when the GPS updates. Simulations have shown that low pass filtering the velocity estimate adds delay that impacts the attitude estimate, and therefore the gravity vector measurement is instead skipped when the GPS provides a measurement to the inertial position and velocity estimator.

One potential issue with differentiating the output of the inertial position and velocity Kalman filter is that it is largely dependent on the GPS. Another method to determine the dynamic acceleration, as described in [28], is to use the Transport Theorem to express the dynamic acceleration as

$$\ddot{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I}} = \frac{d}{dt}^{\mathcal{B}}\left(\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}\right) + \boldsymbol{\omega}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}} \times \mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{I}}. \tag{6.2.37}$$

The second term in 6.2.37 may be calculated by using the angular velocity measurement from the gyroscope, corrected with the bias estimate, as well as the airspeed in place of the inertial velocity. The first term is the rate of change of the inetial velocity relative to the body frame, expressed in body frame coordinates. This term is, however, difficult to obtain, as it requires an accurate model of the aerodynamic drag, and the thrust force produced once the engine has started [28]. This method has not been considered further in this work.

Velocity Vector Measurement

Another inertial vector considered for use in the MEKF is the booster's velocity vector. The kinematic estimator provides an estimate of the velocity vector in inertial coordinates, thus

$$\boldsymbol{\nu}^{\mathcal{I}}_{vel,k} = \frac{\hat{\dot{\mathbf{r}}}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I},k}}{\|\hat{\dot{\mathbf{r}}}^{\mathcal{I}}_{\mathcal{B}/\mathcal{I},k}\|}. \tag{6.2.38}$$

A measurement of the velocity vector in body coordinates would be obtained from the airspeed and wind angles sensors. The measured vector in body coordinates is then

$$\boldsymbol{\nu}^{\mathcal{B}}_{vel,k} = \frac{\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A}}}{\|\mathbf{v}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A}}\|}. \tag{6.2.39}$$

Note that this assumes the air is stationary (i.e. $\mathbf{v}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I}} = 0$ in Equation 3.2.6). Direct measurements of the airspeed and wind angles in the body frame will be influenced by any winds, resulting in noise on the body frame measurement. Wind gusts may be handled as noise by the MEKF, however static winds will cause a consistent bias in the measurements. Simulations have shown that this bias can be significant, therefore the velocity vector is not used directly as a measurement in the attitude estimator.

6.2.3 Wind Angle Observer

Although the wind angle sensors are available to provide direct measurements of the angle of attack and sideslip angle, the sideslip angle is only required for feedback during the oblique wing case. Since this represents a very small portion of the entire flight, an estimator has been investigated to obtain the wind velocity vector from which the sideslip angle can be computed. An estimator proposed by Johansen, et. al, [33] uses the inertial velocity and attitude estimates and a pitot-static probe fixed to the x-axis of the body frame to estimate the wind velocity vector and an offset in the pitot-static probe, ϵ. This estimator is implemented in this work, and briefly described here in notation consistent with the rest of this thesis. The state is the wind velocity vector in inertial coordinates and the pitot-static probe offset, and is assumed to be slowly time-varying. Therefore, the state dynamics are

$$\begin{bmatrix} \dot{\mathbf{v}}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I}} \\ \dot{\epsilon_p} \end{bmatrix} = \mathbf{0}_{4\times 1}. \tag{6.2.40}$$

A measurement of the inertial velocity along the body x-axis in body coordinates is made as

$$V_x = \mathbf{d_1^T} \mathbf{A}^{\mathcal{B}}_{\mathcal{I}} \mathbf{v}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I}} + U_{meas}\epsilon_p, \tag{6.2.41}$$

where $\mathbf{d}_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$. Thus, the measurement matrix is

$$\mathbf{C} = \begin{bmatrix} \mathbf{d}_1 \mathbf{A}^{\mathcal{B}}_{\mathcal{I}} & U_{meas} \end{bmatrix}. \tag{6.2.42}$$

The observability Gramian is

$$W_o(t_0, t_1) = \int_{t_0}^{t_1} \mathbf{C}^T \mathbf{C}\, dt = \int_{t_0}^{t_1} \begin{bmatrix} \left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}\right]^T \mathbf{d}_1 \mathbf{d}_1^T \mathbf{A}^{\mathcal{B}}_{\mathcal{I}} & U_{meas}\left[\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}\right]^T \mathbf{d}_1 \\ \mathbf{d}_1^T \mathbf{A}^{\mathcal{B}}_{\mathcal{I}} U_{meas} & U^2_{meas}, \end{bmatrix} dt. \tag{6.2.43}$$

since the state matrix is zero. Since $rank(\mathbf{d}_1\mathbf{d}_1^T) = 1$, flight at a constant attitude results in $rank(W_o) = 2$. This implies that in order for the state to be observable, the aircraft must constantly change attitude (pitch and yaw) so that $\mathbf{A}^{\mathcal{B}}_{\mathcal{I}}$ changes from t_0 to t_1. Intuitively, the one component of the wind vector measured by the pitot tube is insufficient to estimate the entire vector. Simply rolling does not change the direction of the probe, thus pitch and yaw motions are required. However, Johansen et al. implemented their estimator on flight data from straight and level flights and found that atmospheric disturbances, such as wind gusts, are sufficient to keep the estimate bounded, although they were unable to verify the accuracy due to the lack of an independent measurement of the wind angles. This estimator has been implemented in

this work as a discrete Kalman Filter with the state dynamics

$$\begin{bmatrix} \dot{\mathbf{v}}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I},k+1} \\ \dot{\epsilon}_{p,k+1} \end{bmatrix} = \mathbf{1}_{4\times4} \begin{bmatrix} \mathbf{v}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I},k} \\ \epsilon_{p,k} \end{bmatrix} + \begin{bmatrix} \mathbf{w}_{wind} \\ w_{pitot} \end{bmatrix}, \tag{6.2.44}$$

where $\mathbf{w}_{wind}$ and w_{pitot} are zero-mean white noise processes on the wind and pitot-static probe offset, respectively. The covariance matrix on the wind noise is $\mathbf{Q}_{wind}$, and the variance on the pitot-static probe noise is σ^2_{pitot}. Thus,

$$\mathbf{w}_{wind} \sim \mathcal{N}(0, \mathbf{Q}_{wind})$$

$$w_{pitot} \sim \mathcal{N}(0, \sigma^2_{pitot}).$$

The covariance matrix for the wind EKF is

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{wind} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & \sigma^2_{pitot} \end{bmatrix}. \tag{6.2.45}$$

The wind EKF is implemented as follows:

1. Compute the measurement matrix using the current attitude estimate and pitot-static probe measurement:

$$u_{k+1} = \mathbf{C}_{k+1}\mathbf{x}_{k+1} = \begin{bmatrix} \mathbf{d}_1 \hat{\mathbf{A}}^{\mathcal{B}}_{\mathcal{I},k+1} & U_{meas,k} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{v}}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I},k+1} \\ \hat{\epsilon}_k \end{bmatrix}$$

2. Propagate the estimate covariance:

$$\mathbf{P}_{k+1|k} = \mathbf{P}_{k|k} + \mathbf{Q} \tag{6.2.46}$$

3. Compute the gain:

$$\mathbf{L}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{C}_k^T(\mathbf{C}_k\mathbf{P}_{k+1|k}\mathbf{C}_k^T + R)^{-1}$$

4. Compute the state estimate:

$$\hat{\mathbf{x}}_{k+1} = \begin{bmatrix} \hat{\mathbf{v}}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I}} \\ \hat{\epsilon} \end{bmatrix}_{k+1} = \hat{\mathbf{x}}_k + \mathbf{L}_{k+1}(u_{k+1} - \mathbf{C}_{k+1}\hat{\mathbf{x}}_k)$$

5. Update the estimate covariance:

$$\mathbf{P}_{k+1|k+1} = (\mathbf{1}_{4\times4} - \mathbf{L}_{k+1}\mathbf{C}_{k+1})\,\mathbf{P}_{k+1|k}\,(\mathbf{1}_{4\times4} - \mathbf{L}_{k+1}\mathbf{C}_{k+1})^T + \mathbf{L}_{k+1}R\mathbf{L}_{k+1}^T$$

6. Compute the booster velocity relative to the air:

$$\hat{\mathbf{v}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A},k+1} = \begin{bmatrix} \hat{U}_{k+1} \\ \hat{V}_{k+1} \\ \hat{W}_{k+1} \end{bmatrix} = \hat{\mathbf{A}}^{\mathcal{B}}_{\mathcal{I},k+1}\left(\hat{\mathbf{r}}^{\mathcal{I}}_{\mathcal{B},k+1} - \hat{\mathbf{v}}^{\mathcal{I}}_{\mathcal{A}/\mathcal{I},k+1}\right)$$

7. Compute the airspeed and wind angles:

$$\hat{V}_{\infty,k+1} = \|\hat{\mathbf{v}}^{\mathcal{B}}_{\mathcal{B}/\mathcal{A},k+1}\|$$

$$\hat{\alpha}_{k+1} = \tan^{-1}\left(\frac{\hat{U}_{k+1}}{\hat{W}_{k+1}}\right)$$

$$\hat{\beta}_{k+1} = \sin^{-1}\left(\frac{\hat{V}_{k+1}}{\hat{V}_{\infty,k+1}}\right)$$

This wind velocity estimator is not ideal for use in the return flight, as it requires significant variations in the booster's attitude to be fully observable. However, the majority of the return flight will be a constant heading. Simulations have verified that components of wind not aligned with the booster's $\bar{x}_{\mathcal{B}}$-axis are not observable. Bank reversals were considered in the rocket configuration to increase the observability of this estimator, however the booster does produce sufficient lift to achieve large enough changes in heading to significantly increase the observability of the wing velocity estimator. This verifies the initial intuition that sensors will be required to directly measure the wind angles at altitudes at which atmospheric winds are significant.

6.3 Conclusion

This chapter has developed the estimators used to obtain measurements of the states required for feedback. A Kalman filter is used to to obtain an estimate of the inertial position and velocity, from which the booster's heading may be derived. This filter uses the accelerometer to propagate the estimate and updates the state with GPS measurements, which contain significant delays. A multiplicative extended Kalman filter is used to estimate the booster's attitude. The inertial reference vectors considered are Earth's magnetic field, gravity field and the booster's velocity vector. Dynamic acceleration must be removed from the accelerometer measurements in order to measure the gravity field, which is performed by differentiating the output of the inertial position and velocity estimator. Finally, a wind velocity estimator has been

considered, which utilises a measurement from a pitot-static probe and the inertial velocity and attitude estimates to estimate the airspeed, angle of attack and sideslip angle.

Chapter 7

Simulation Results

This chapter introduces the simulation environment used to verify the controller and estimator designs and provides a comparison of the results.

7.1 Simulation Environment

A six degree-of-freedom (DOF) simulation environment is developed in Simulink using the nonlinear rigid body equations of motion, and the aerodynamic forces and moments model, developed in Chapter 3. Figure 7.1 shows the top level Simulink blocks.

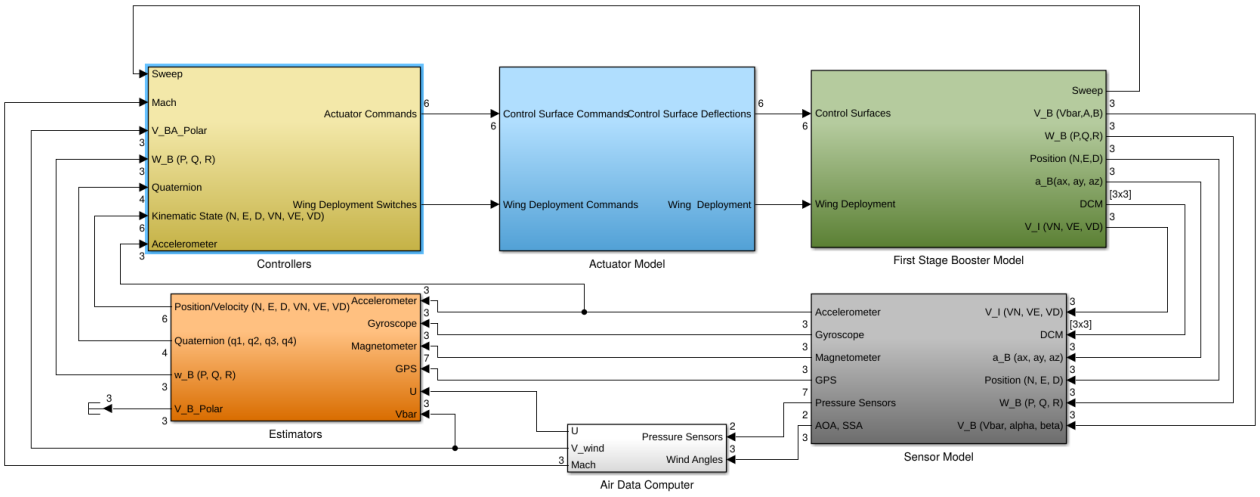


Figure 7.1: Top level block diagram of Simulink Model.

7.2 Controller Results and Comparison

This section presents the results of the control system design and justifies the design decisions made. The nominal trajectory used to compare the controllers starts at an initial condition with an altitude of 10 km and airspeed of Mach 0.8. The booster is assumed to initially be in an equilibrium condition at this altitude, as determined by the algorithm in Section 3.5.2. All of the figures in this section show the

gain schedule numbers on the plots. Figure 7.2 shows the wing sweep angle, Mach number and altitude throughout the descent trajectory. The wing reaches a sweep angle of $\Lambda = 60°$ at an altitude of approximately 6 km, and regains approximately 1 km of altitude in the pull-up manoeuvre before settling into the minimum sink rate glide at approximately Mach 0.15.

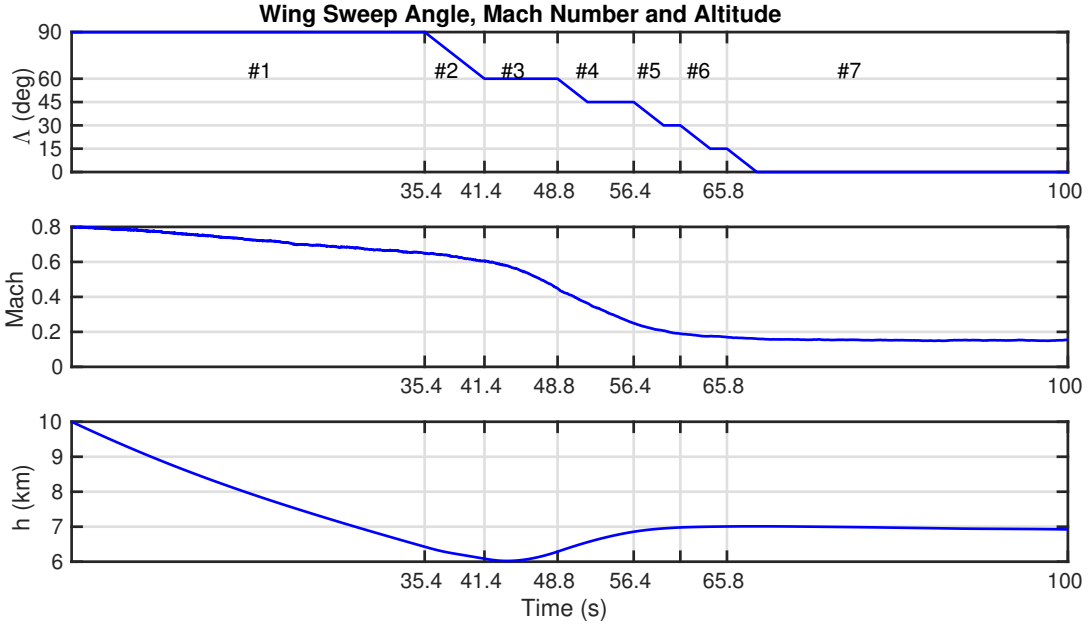


Figure 7.2: Wing sweep angle, Mach number and altitude for the nominal control design.

Figure 7.3 shows the roll, pitch and heading angles throughout the descent trajectory. The trim pitch angle is $-26.6°$, and is used in the attitude controller to maintain constant altitude during the initial wing deployment (gain schedules #2). Once the wing reaches $\Lambda = 60°$, the airspeed controller is activated to pull-up and then achieve the minimum sink rate glide by providing a pitch angle reference to the attitude controller. Figure 7.4 shows the polar velocity throughout the descent trajectory. The trim angle of attack is 6.85°, and a reference of 8.5° is provided to the attitude controller in the rocket configuration (gain schedule #1) to begin pulling up. The angle of attack limits of the wing's airfoil is $-5°$ to 10°, and the transition between the pull-up manoeuvre and the minimum sink rate glide has been designed to remain within these limits. A constant 0° sideslip reference is provided to the attitude controller throughout the wing deployment. The primary disturbances to the sideslip angle, prior to the activation of the heading angle controller (gain schedule #7), result from the rotation of the wing. The non-constant roll angle reference from the heading angle controller causes slightly larger disturbances in sideslip, however the sideslip angle remains within $\pm 2°$ throughout the wing deployment.

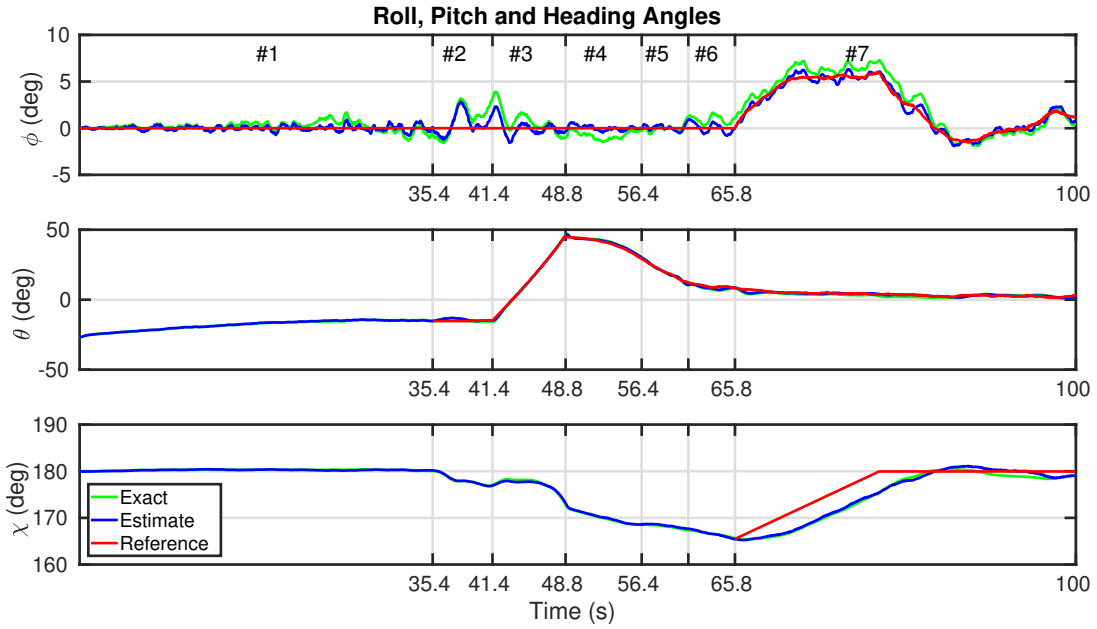


Figure 7.3: Roll, pitch and heading angles for the nominal control design.

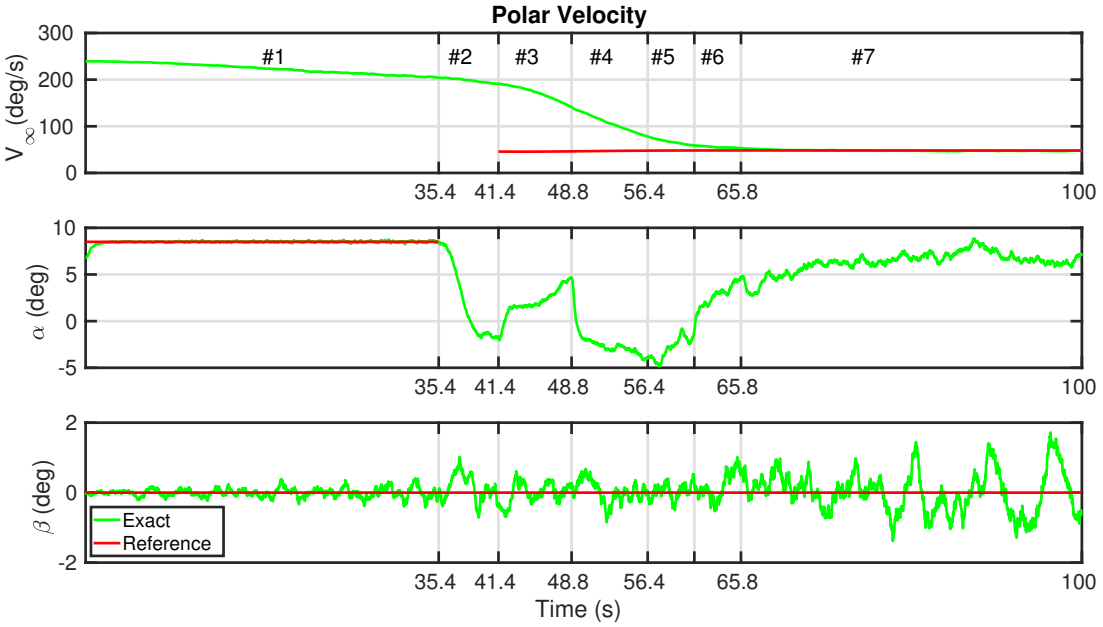


Figure 7.4: Polar velocity (airspeed, angle of attack and sideslip angle) for the nominal control design.

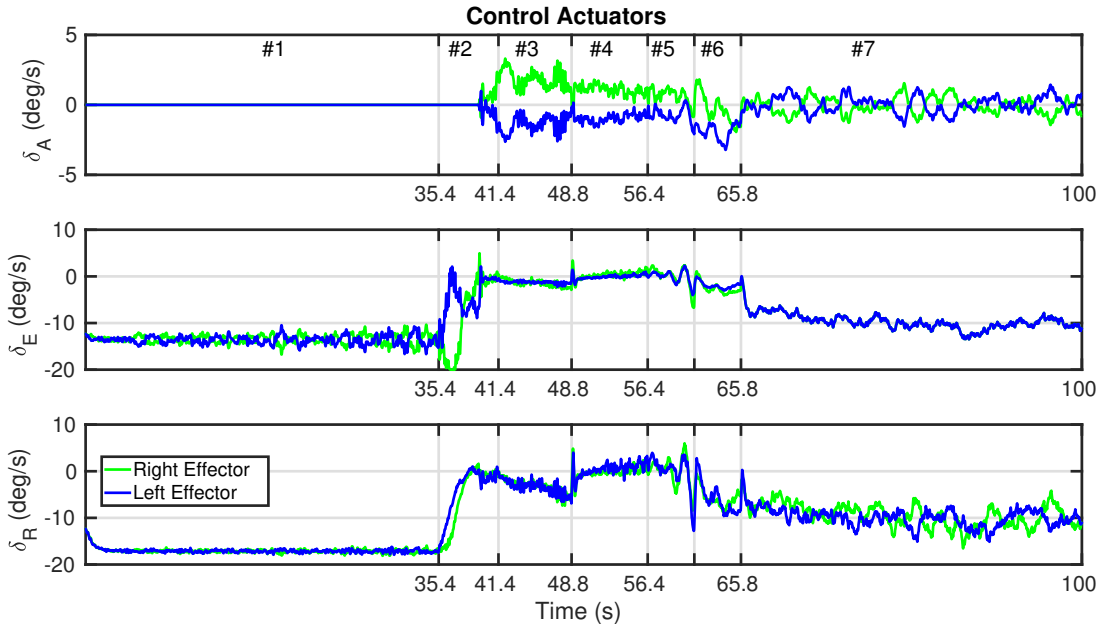


Figure 7.5: Aerodynamic control surface deflections for the nominal control design.

The remainder of this section investigates the various design decisions that resulted in this nominal trajectory. Specifically, the explicit ganging and the redistributed pseudo inverse control allocation algorithms are first compared. Next, the pitch angle and flight path angle are compared as the states that are rate limited to remain within the booster's load factor limits during the pull-up manoeuvre. Then, two nonlinear functions, the *fal* function and the hyperbolic tangent function, are compared for their use as the airspeed controller's proportional term.

7.2.1 Atmospheric Wind Modelling

In order to model external disturbances, two atmospheric wind models are used. The first is the Horizontal Wind Model (HWM) [60], which models the average daily horizontal winds at a given latitude, longitude and altitude. The second model is the Dryden model, which models turbulence and wind gusts. Figure 7.6 shows the static wind gusts from the HWM over Stellenbosch, South Africa, along with wind gusts from the Dryden model throughout the descent trajectory. Stellenbosch was chosen for the simulation since the jet stream at altitudes of approximately 10 km results in significant winds in the east-west direction. Figure 7.7 shows the angular velocity in body coordinates resulting from the Dryden model for light turbulence with a probability of exceedance intensity of 10^{-2}.

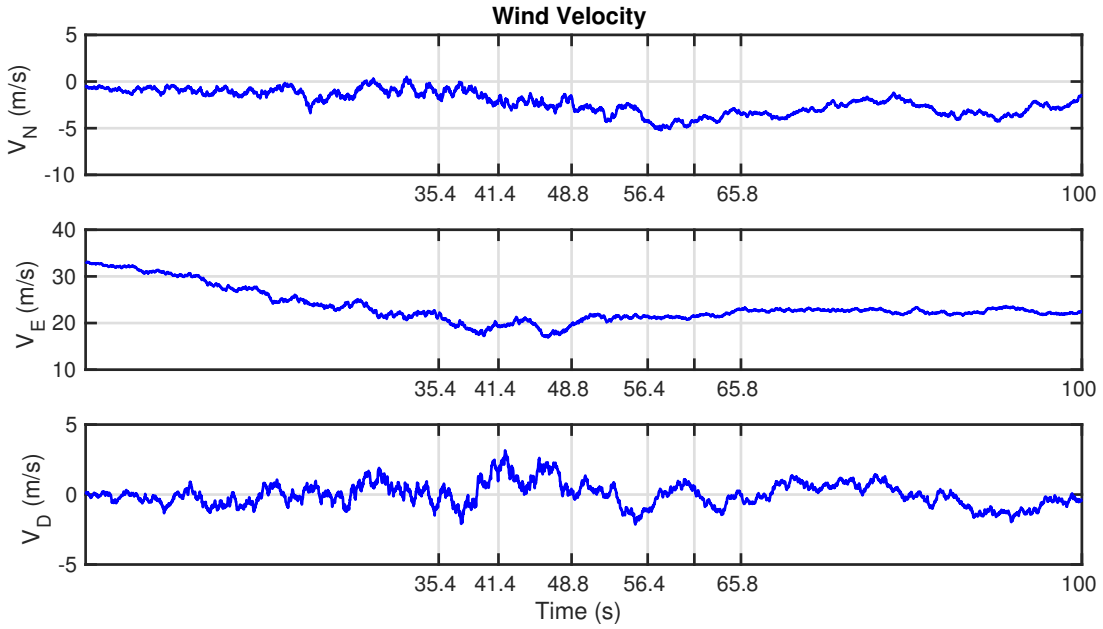


Figure 7.6: Static winds from the Horizontal Wind Model for Stellenbosch, South Africa.

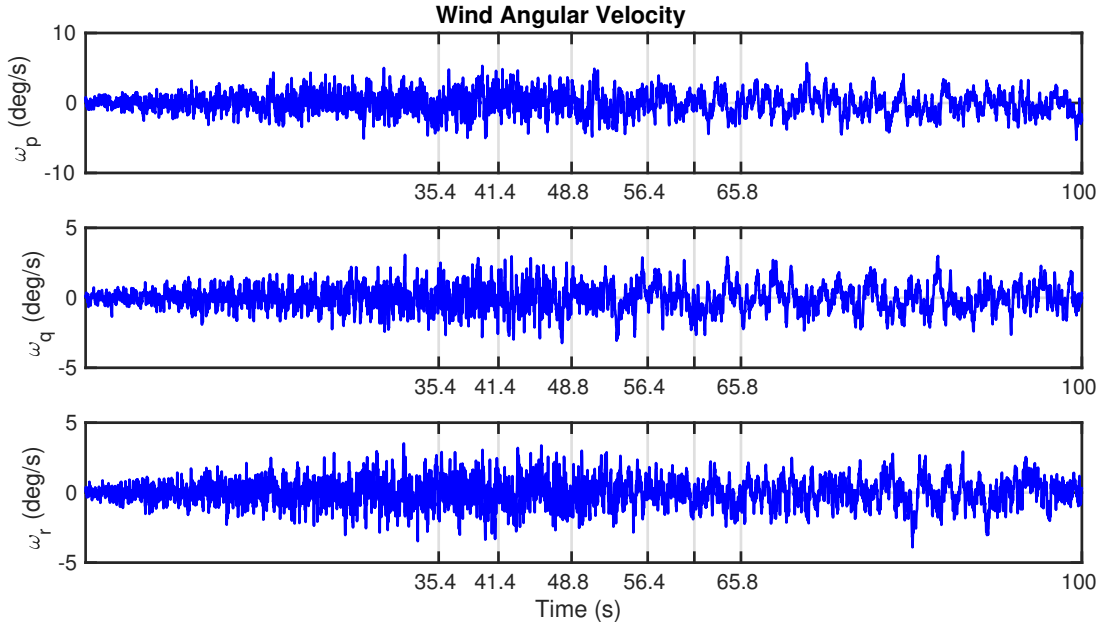


Figure 7.7: Dryden model of light turbulence (probability of exceedance intensity of 10^{-2})

7.2.2 Control Allocation

The primary purpose of the recursive pseudo inverse algorithm described in Section 5.2.1 is to handle the case when the control surfaces saturate, which is expected to occur during the initial wing deployment in the presence of significant external disturbances. To compare the explicit ganging with the pseudo inverse control allocation, the HWM and turbulence models provide the external disturbances, and the resulting control effector deflections and roll angle are shown in Figure 7.8 for explicit ganging and Figure 7.9 for redistributive pseudo inverse. For both cases, the right elevon saturates shortly after deploying the wing.

When using explicit ganging control allocation, the control allocation algorithm is unable to recognize this saturation and the outer loop feedback controllers are required to compensate for the remaining moment required to maintain zero roll angle. With the redistributive pseudo inverse, we see that the algorithm immediately recognizes that additional roll moment is required and commands the left elevon to produce a positive roll moment to return to zero roll angle.

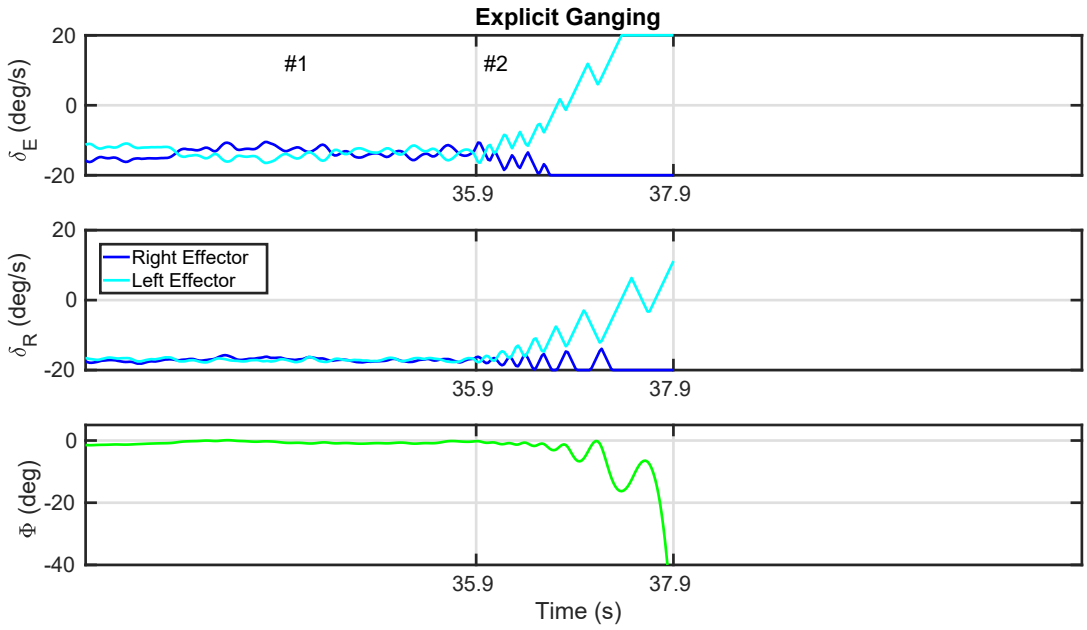


Figure 7.8: Elevon and ruddervator deflections during the initial wing deployment using explicit ganging control allocation.

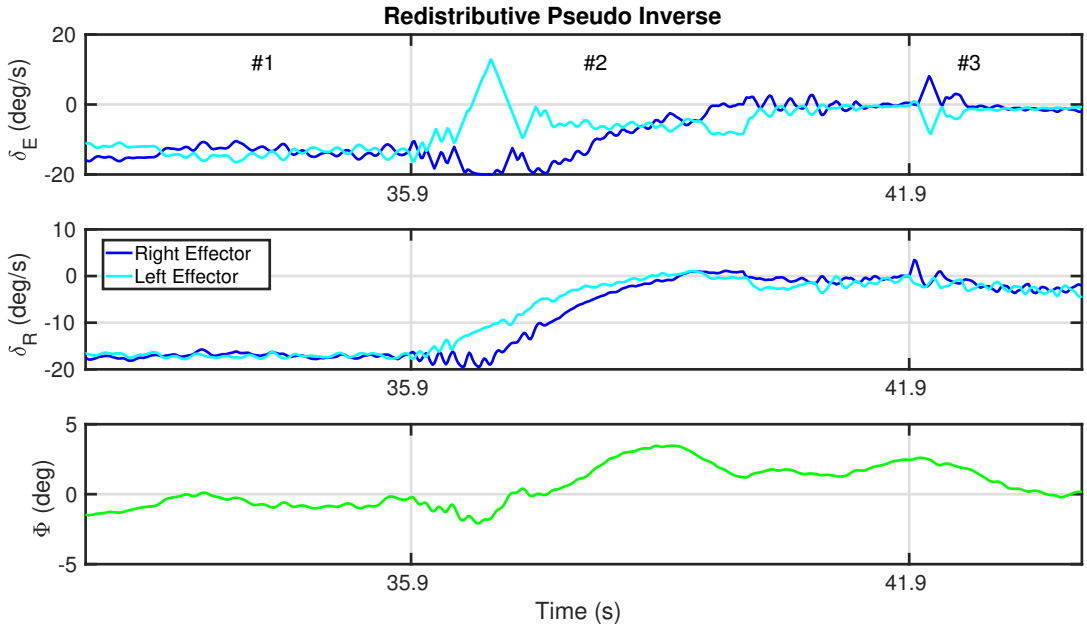


Figure 7.9: Elevon and ruddervator deflections during the initial wing deployment using recursive pseudo inverse control allocation.

One issue with the pseudo inverse control allocation method is that the solution is not unique, since

the control allocation problem in Equation 5.2.2 is underdetermined. This may result in a situation where the required moment is not ideally distributed to the control surfaces. For instance, Figure 7.10 shows the control surface deflections when the redistributed pseudo inverse control allocation is used for the entire simulation. When the wing is fully deployed (gain schedule #7), the angular velocity controller commands a positive pitching moment. The resulting pseudo inverse allocates some of this moment to the ailerons, resulting in a offset around a deflection of $-3°$ for both the left and the right aileron during the minimum sink rate glide. Using the ailerons to achieve pitching moment in the aircraft configuration is not ideal. A negative aileron command results in an upwards deflection, which causes the wing to produce less lift. Therefore, the booster must fly at larger angle of attack to achieve the minimum sink rate glide, as shown in Figure 7.11.

Various methods, such as linear programming or mixed optimisation [49], are available to drive the control effectors to a desired position if there is excessive control authority available to achieve the desired moment. However, this is overcomplicated in this instance, since the effect of each group of control surfaces is symmetrical in the aircraft configuration. Therefore, the nominal control surface deflections, shown in Figure 7.5, switches back to explicit ganging for gain schedule #7 leaving the ailerons to only be used to achieve the desired roll moment. Future work is required to verify that the ailerons have sufficient control authority to reject the roll moment disturbance resulting from starting the piston engine and deploying the propeller.

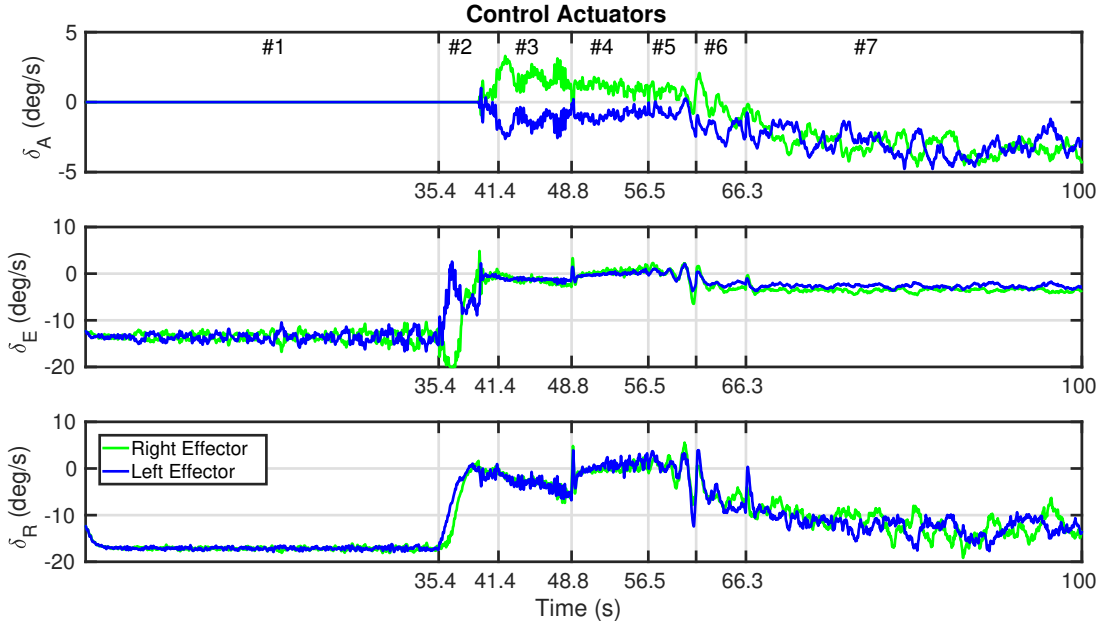


Figure 7.10: Control surface deflections when using the pseudo inverse control allocation in the aircraft configuration.

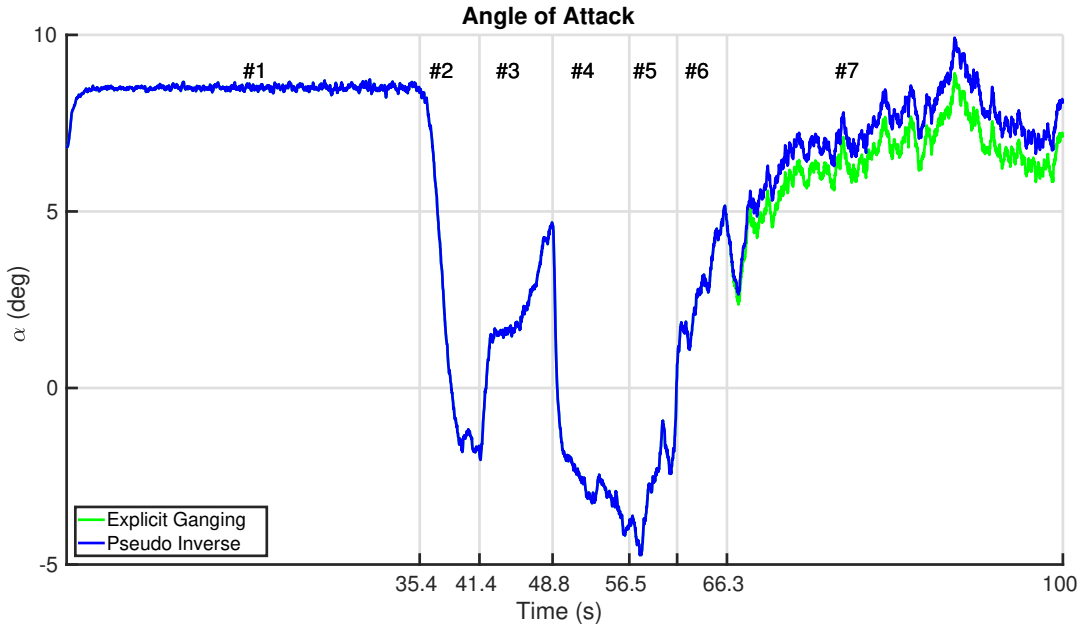


Figure 7.11: Control surface deflections when using the pseudo inverse control allocation in the aircraft configuration.

7.2.3 Pull-up Manoeuvre

The two options investigated to perform the pull-up manoeuvre while respecting the load factor of the booster are to rate limit the pitch angle or flight path angle, as described in Section 5.3.2. The maximum load factor n_{max} is selected as 3.8, as this is the minimum load factor aircraft must be certified for to meet governmental airworthiness requirements, such as [61]. Figure 7.12 shows the pitch angle and angle of attack for both options, and Figure 7.13 shows the load factor and altitude gain during the pull-up manoeuvre. The blue line in both figures shows the results when rate limiting the flight path angle, and the magenta and green lines show the results when rate limiting the pitch angle. In Figure 7.12, the red line shows the reference angle of attack commanded by the flight path angle controller and the reference given to the attitude controller when commanding the pitch angle. Interestingly, the resulting pull-up manoeuvre for both methods are nearly identical, with the pitch angle feedback achieving a slightly faster response. This shows that the assumption that the rate of change of the angle of attack is negligible when considering the load factor is reasonable. The response is likely faster for the pitch angle feedback since the flight path angle controller adds an additional feedback loop, adding a phase delay. Therefore, the pitch angle is selected as the state to be rate limited during the pull-up manoeuvre to remain within the booster's load factor limits. During the initiation of the pull-up manoeuvre for both methods, the load factor slightly exceeds $n_{max} = 3.8$. The load factor equation was derived for a steady pull-up manoeuvre, and therefore factor of safety must be applied to the value of n_{max} provided to the rate limiters. A value of $n_{max} = 3.5$ was found through simulation to be sufficient when rate limiting the pitch angle, as shown by the green line in Figure 7.13.

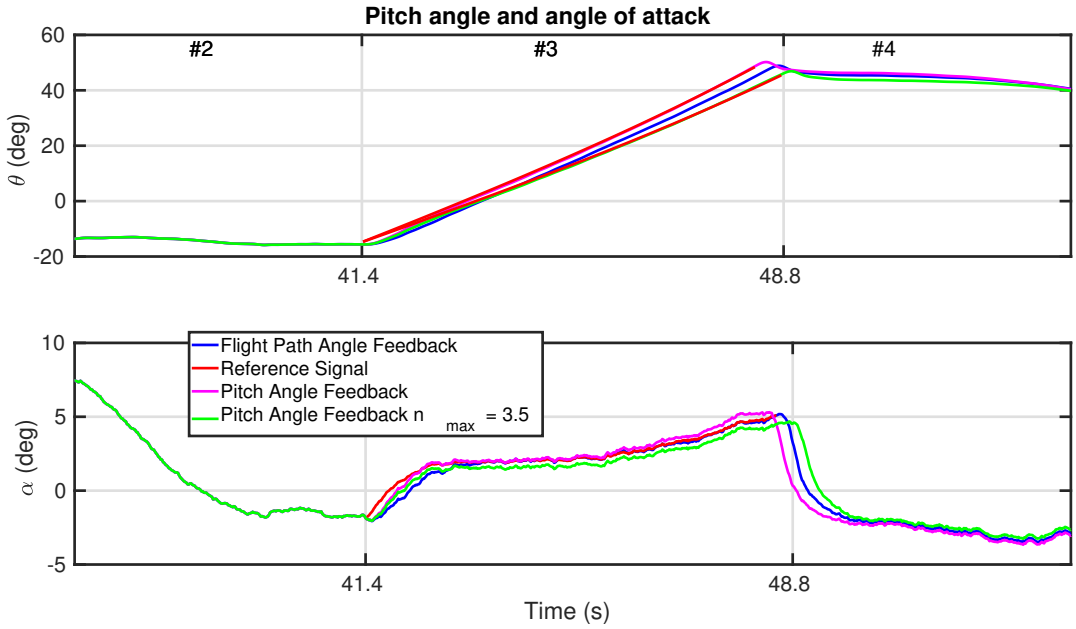


Figure 7.12: Pitch angle and angle of attack during the pull-up manoeuvre.

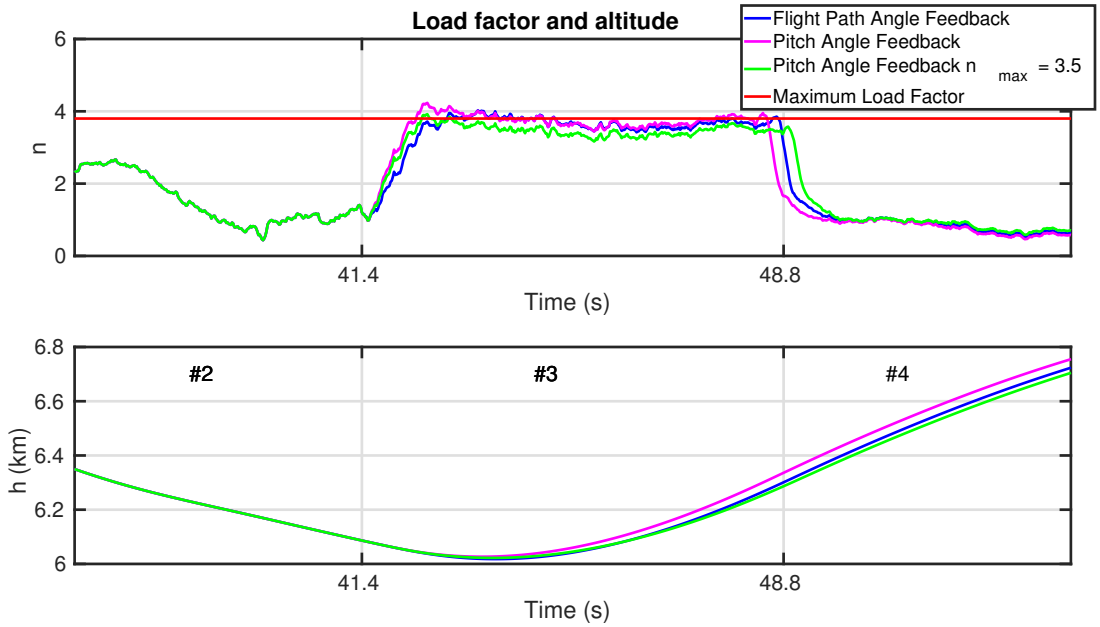


Figure 7.13: Load factor and altitude gain during the pull-up manoeuvre.

7.2.4 Minimum Sink Rate Glide

Once the booster has performed the pull-up manoeuvre, it must then attain the minimum sink rate glide. A proportional-integral controller is implemented on the airspeed to achieve this, with the proportional term being nonlinear to saturate at large airspeed errors and reduce to a linear proportional controller at small terms. The two nonlinear terms considered are the *fal* function proposed by [52] and the hyperbolic tangent function proposed in this work. The pitch angle command and airspeed reference are shown in Figure 7.14, and the resulting angle of attack and altitude are shown in Figure 7.15.

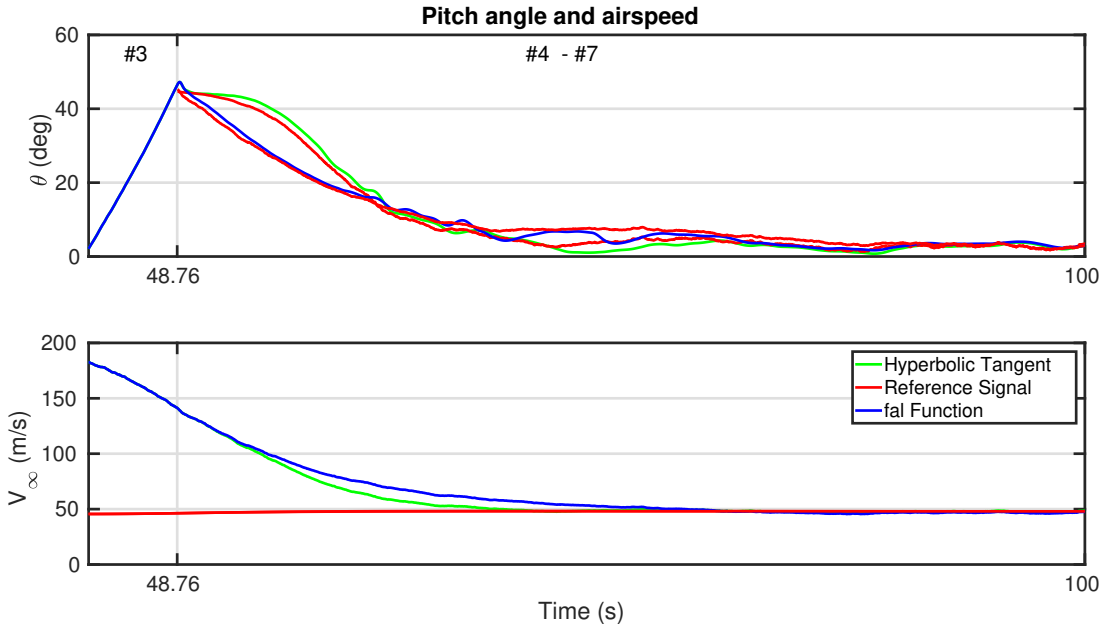


Figure 7.14: Pitch angle and airspeed during the minimum sink rate glide.

The hyperbolic tangent function obtains the minimum sink rate glide both faster and at a higher altitude than the *fal* function. This is due to the fact that, when the minimum sink rate glide control is initiated, the booster is at a pitch angle of 49.7° and the airspeed error is $94.9 m/s$. Thus, from Equation 5.3.11, the value of a_2 in the *fal* function is 0.858. This results in a nearly linear control law, and shows the hyperbolic tangent function is advantageous in this instance.

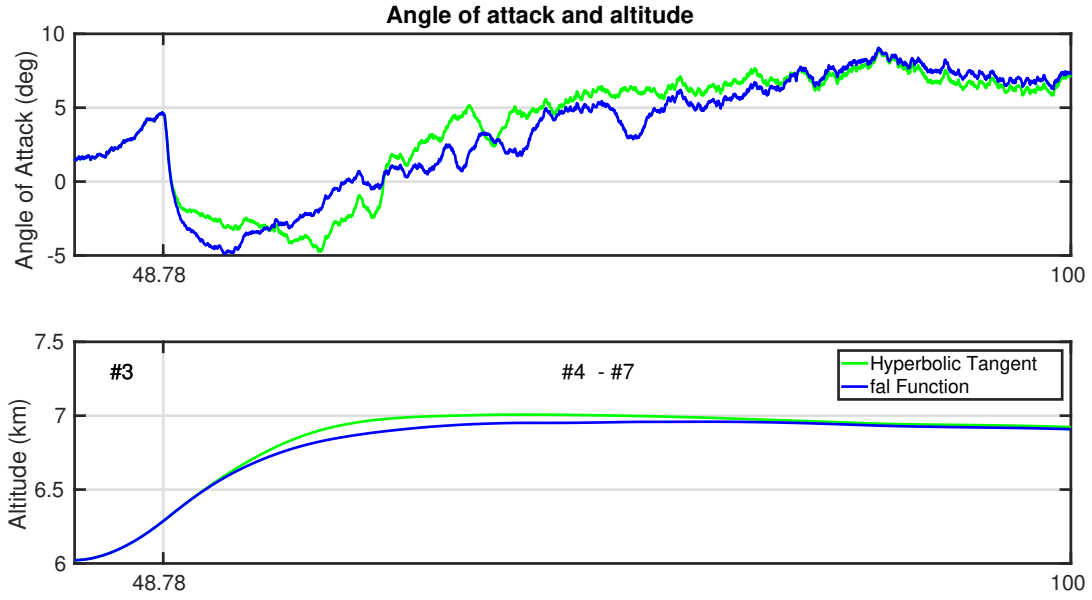


Figure 7.15: Angle of attack and altitude during the minimum sink rate glide.

Throughout this transition from the end of the pull-up manoeuvre to the minimum sink-rate glide, the booster must pitch down in order to maintain near level flight. This causes the angle of attack to be

very small. The wing cannot be completely deployed from $\Lambda = 60°$ to $\Lambda = 0°$ at the end of the pull-up manoeuvre, since fully deployed wing generates significantly more lift at this high velocity. Therefore, the Mach numbers at which the booster transitions between gain schedule numbers #4 and #7 were iteratively chosen so that the booster remains within the wing's angle of attack limits between $-5°$ and $10°$ and attains the minimum sink rate glide with the wing fully deployed.

The problem of determining this transition from the pull-up manoeuvre to the minimum sink rate glide could be interestingly formulated as a trajectory optimisation problem. The initial condition would be the state of the booster after first partially deploying the wing to $\Lambda = 60°$, and the final condition would be the minimum sink rate glide with the wing fully deployed. The trajectory optimisation would then determine the optimal state trajectory to deploy the wing and pull up while remaining within the load factor and angle of attack limitations of the booster. The development and implementation of such an algorithm is not within the scope of this thesis however, and is left for future work.

7.2.5 Heading Angle Controller

The heading angle controller is tasked with maintaining a constant heading angle during the minimum sink rate glide. Figure 7.16 shows the booster's roll, pitch and heading angles when the heading angle controller is activated at $\Lambda = 60°$. The heading angle is not tracked well until the booster slows down to the airspeed for the minimum sink rate glide, although the total heading disturbance is less than when the booster is allowed to drift prior to $\Lambda = 0°$, as shown in Figure 7.3. The gain on the heading angle controller could be increased to provide better tracking in the oblique wing configuration. However, this would then command larger roll angles in the oblique wing configuration, which contradicts with the primary objective of reducing the velocity to fully deploy the wing. Furthermore, the purpose of the heading angle controller is to prevent the possibility of the multiple boosters colliding while they are starting the engines. Since the pull-up manoeuvre is very fast, the booster's should not be close enough to each other such that random motion will cause an issue. Therefore, the heading angle controller is only implemented after the wing has fully deployed and the booster is gliding prior to starting the engine.

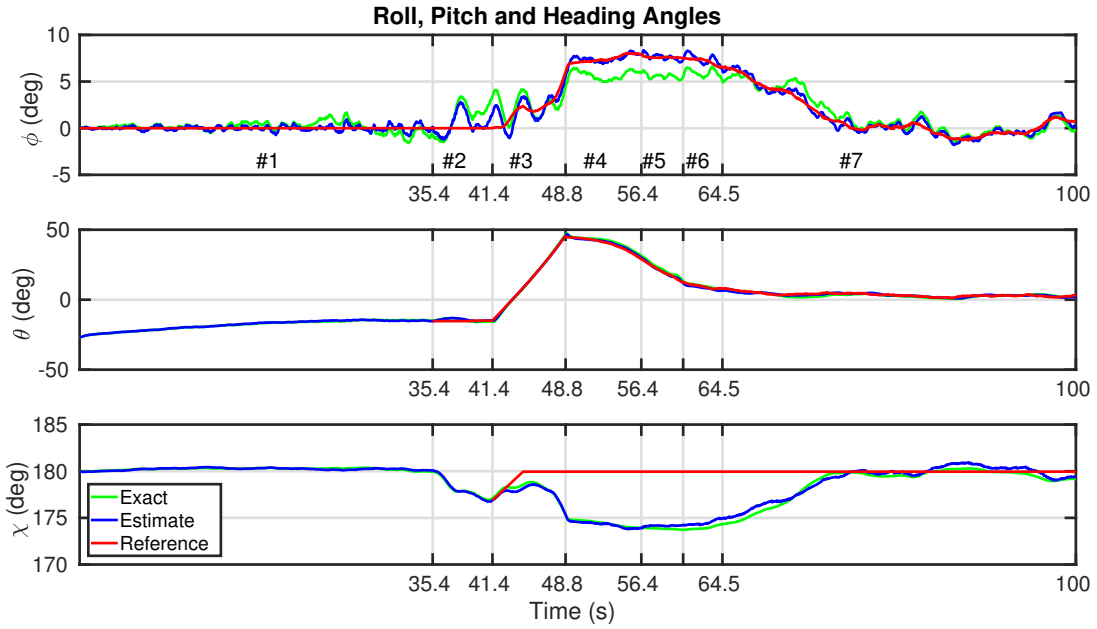


Figure 7.16: Roll angle, pitch angle and heading angle when the heading angle controller is activated at a sweep angle of 60°.

7.3 Estimator Results and Comparison

This section shows the simulation results for the estimator designs. The three inertial reference vectors considered for the attitude multiplicative extended Kalman filter (MEKF) are compared. Furthermore, the handling of the measurement delay on the GPS in the position and velocity Kalman filter is considered. The estimate error figures for all scenarios are provided in this section.

7.3.1 Attitude Multiplicative Extended Kalman Filter

The attitude MEKF uses measurements of inertial vectors in the body frame to estimate the attitude of the booster. The inertial vectors considered are Earth's magnetic field, Earth's gravity field and the booster's velocity vector.

Magnetic Field Measurement

The attitude MEKF requires two reference vectors in order to be observable. This is exemplified in Figure 7.17, which shows the attitude estimate using the magnetic field as the only measurement. Figure 7.18 shows the attitude estimate error.

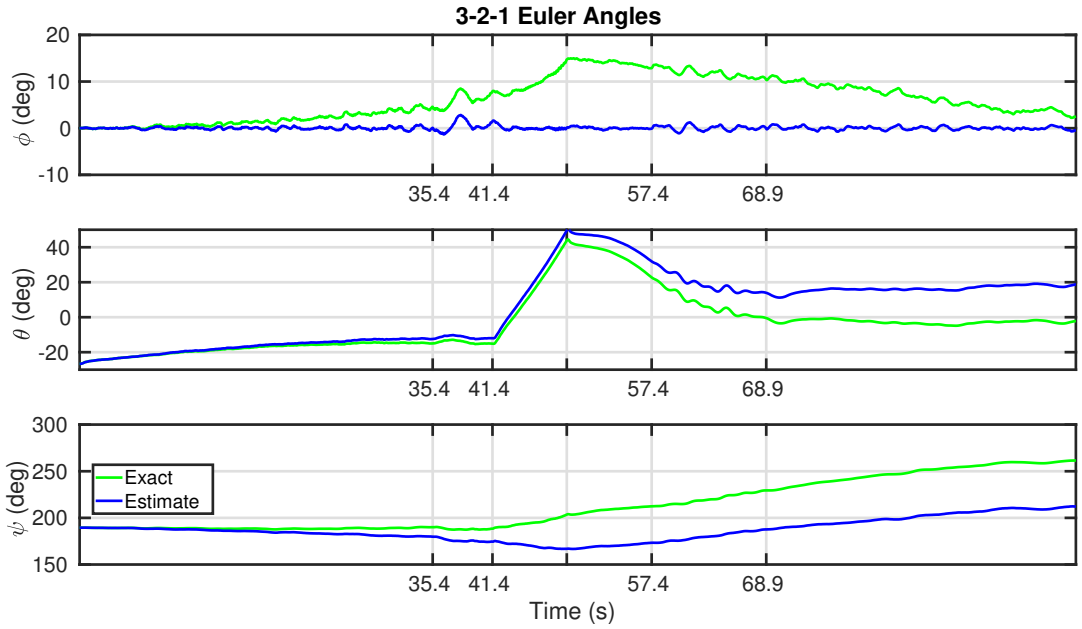


Figure 7.17: Attitude estimate with magnetic field measurements.

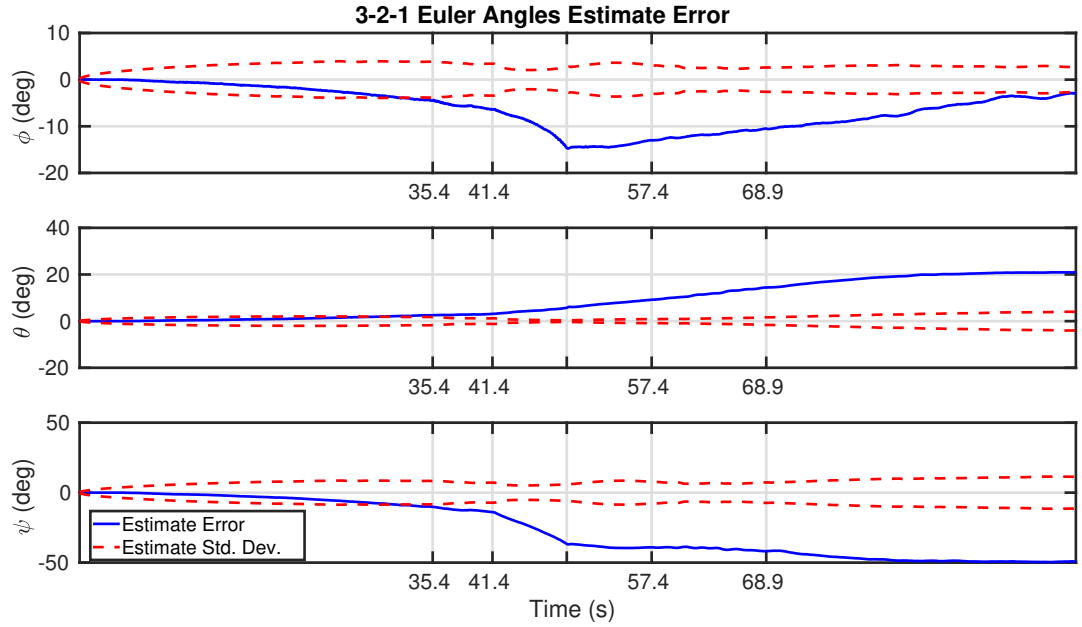


Figure 7.18: Attitude estimate error with magnetic field measurements.

Figure 7.19 shows the gyroscope bias estimate using the magnetic field as the only measurement, and Figure 7.20 shows the estimate error.

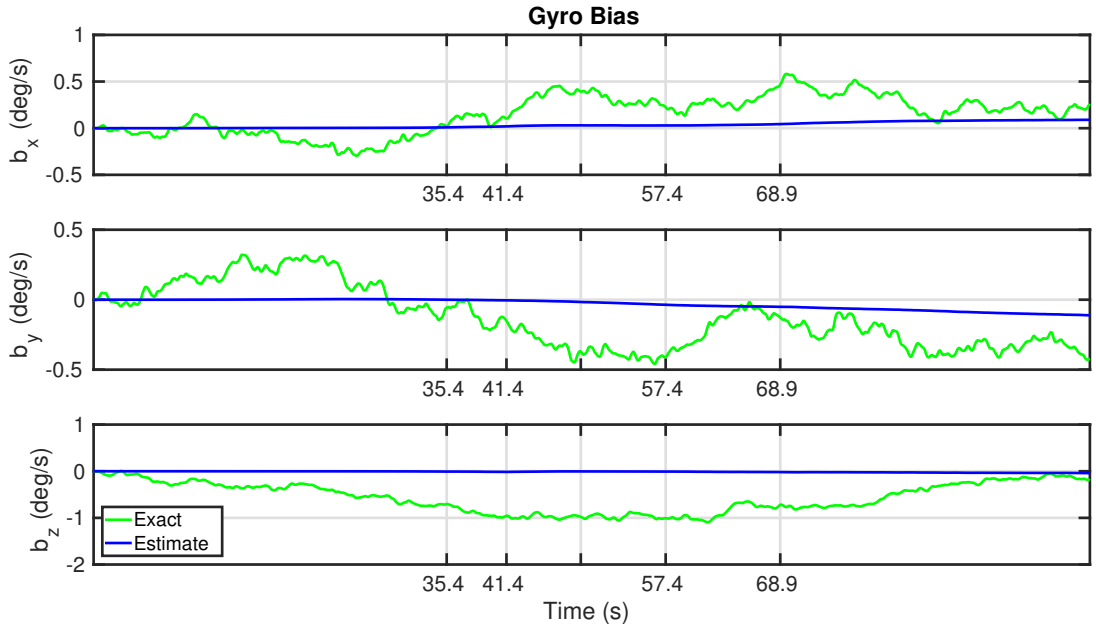


Figure 7.19: Gyroscope bias estimate with magnetic field measurements.

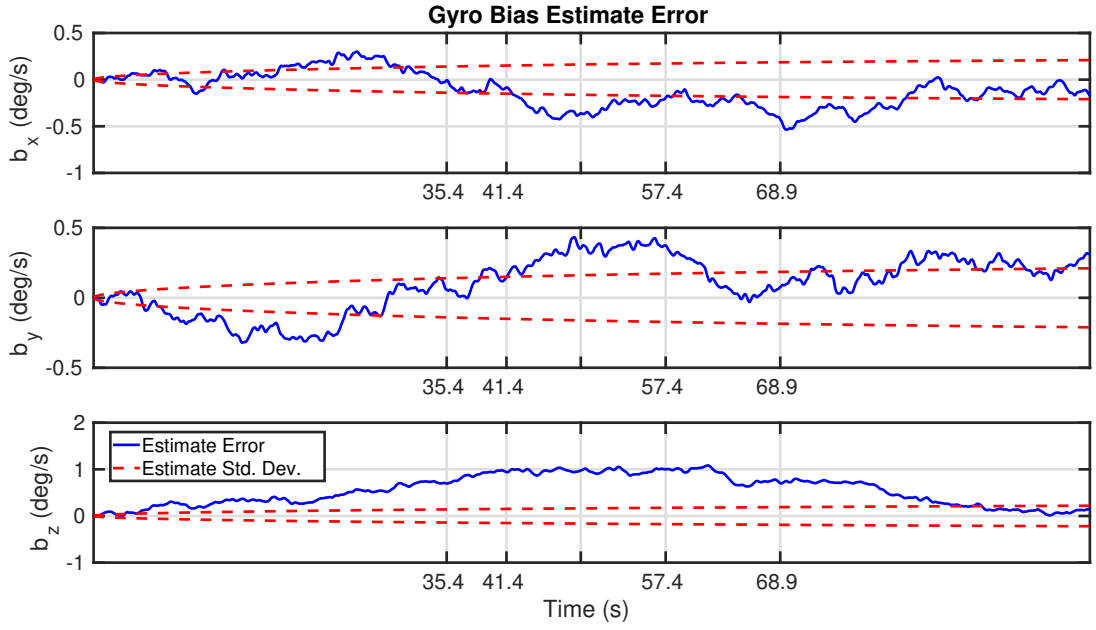


Figure 7.20: Gyroscope bias estimate error with magnetic field measurements.

The estimation error statistics are given in Table 7.1.

Table 7.1: Estimate errors for magnetic field measurements.

Estimation Error	RMS Error	Standard Deviation	Max Error
Roll Angle (deg)	7.675	4.486	14.78
Pitch Angle (deg)	11.68	7.787	20.94
Yaw Angle (deg)	32.84	19.40	49.66
Bias x-axis (deg/s)	0.2201	0.1878	0.5357
Bias y-axis (deg/s)	0.2188	0.1959	0.4333
Bias z-axis (deg/s)	0.6396	0.3256	1.084

Magnetic field measurements alone result in significant errors throughout the wing deployment, and the gyroscope bias estimate remains near zero. This result is expected, as two reference vectors are required for the attitude estimator to be observable, and additional inertial reference vectors are required.

Magnetic and Gravitational Fields

Measurements of the gravity vector are first considered to supplement the magnetic field measurement. Figure 7.21 shows the attitude estimate using measurements of Earth's magnetic and gravity fields. Figure 7.22 shows the attitude estimate error.

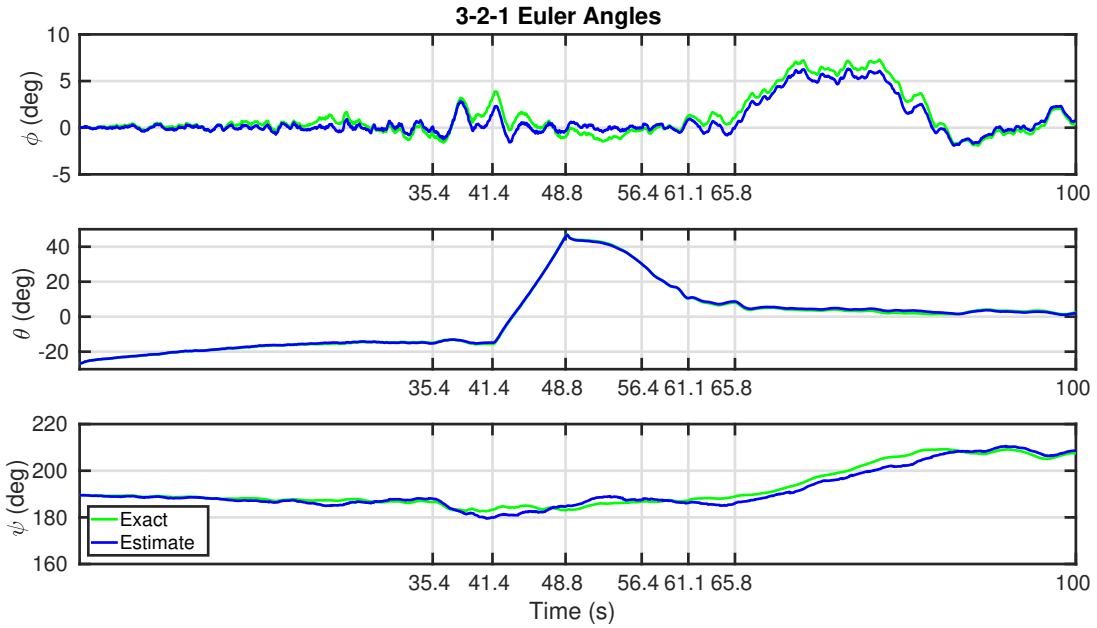


Figure 7.21: Attitude estimate with magnetic field and gravity measurements.

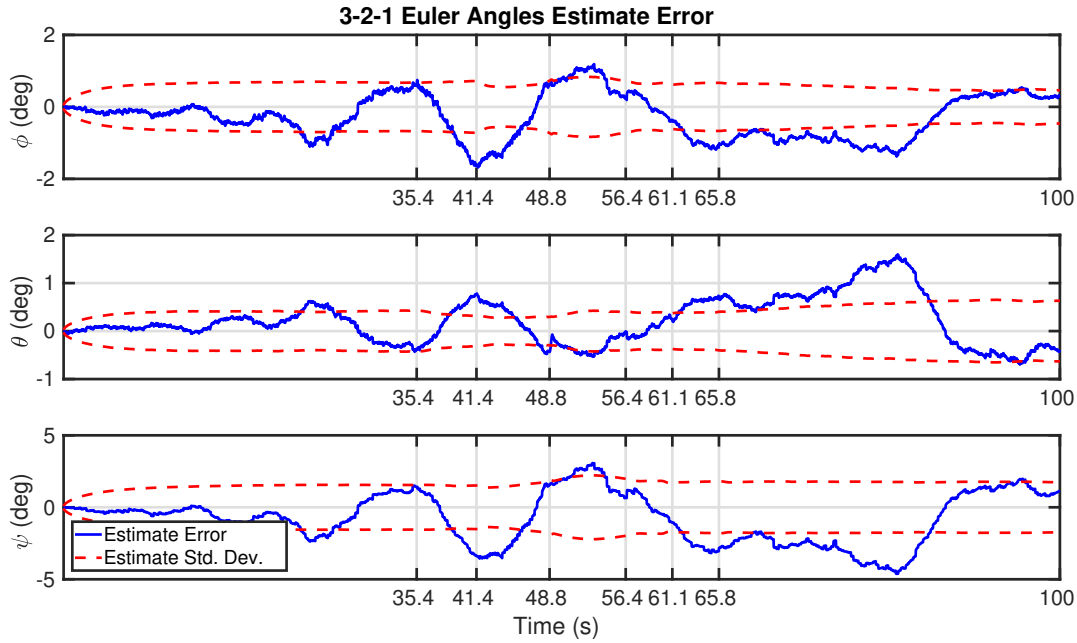


Figure 7.22: Attitude estimate error with magnetic field and gravity measurements.

Figure 7.23 shows the gyroscope bias estimate using the magnetic field and gravity vector measurements, and Figure 7.24 shows the estimate error.

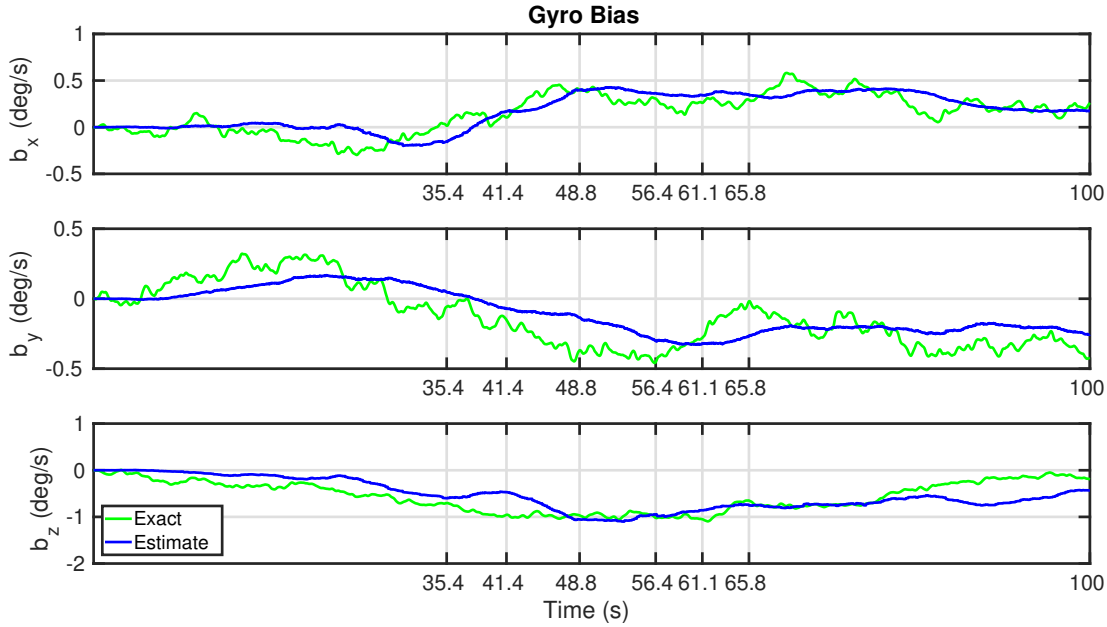


Figure 7.23: Gyroscope bias estimate with magnetic field and gravity measurements.

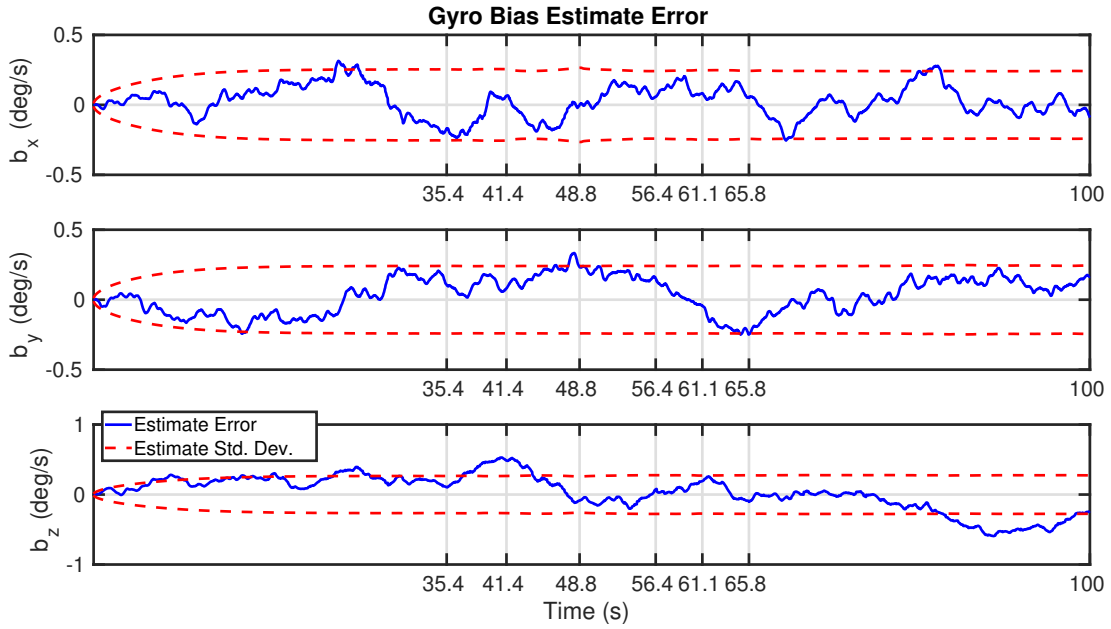


Figure 7.24: Gyroscope bias estimate error with magnetic field and gravity measurements.

The estimation error statistics are given in Table 7.2.

Table 7.2: Estimate errors for magnetic field and gravity vector measurements.

Estimation Error	RMS Error	Standard Deviation	Max Error
Roll Angle (deg)	0.6859	0.6225	1.683
Pitch Angle (deg)	0.5427	0.4966	1.593
Yaw Angle (deg)	1.945	1.780	4.622
Bias x-axis (deg/s)	0.1175	0.1143	0.3139
Bias y-axis (deg/s)	0.1345	0.1304	0.3335
Bias z-axis (deg/s)	0.2549	0.2531	0.5929

Using the gravity vector as a measurement significantly improves the attitude estimate, and the estimate remains accurate throughout the simulation.

Magnetic Field and Velocity Vector

The measurement of the gravity vector is obtained by removing the dynamic acceleration of the booster from accelerometer measurements. The dynamic acceleration is obtained by differentiating the output of the inertial position and velocity Kalman filter, which relies heavily on the GPS. To potentially remove this dependence on the GPS, the velocity vector was also considered to supplement the magnetic field measurement. Figure 7.25 shows the attitude estimate using the magnetometer and the velocity vector for measurements. Figure 7.26 shows the attitude estimate error.

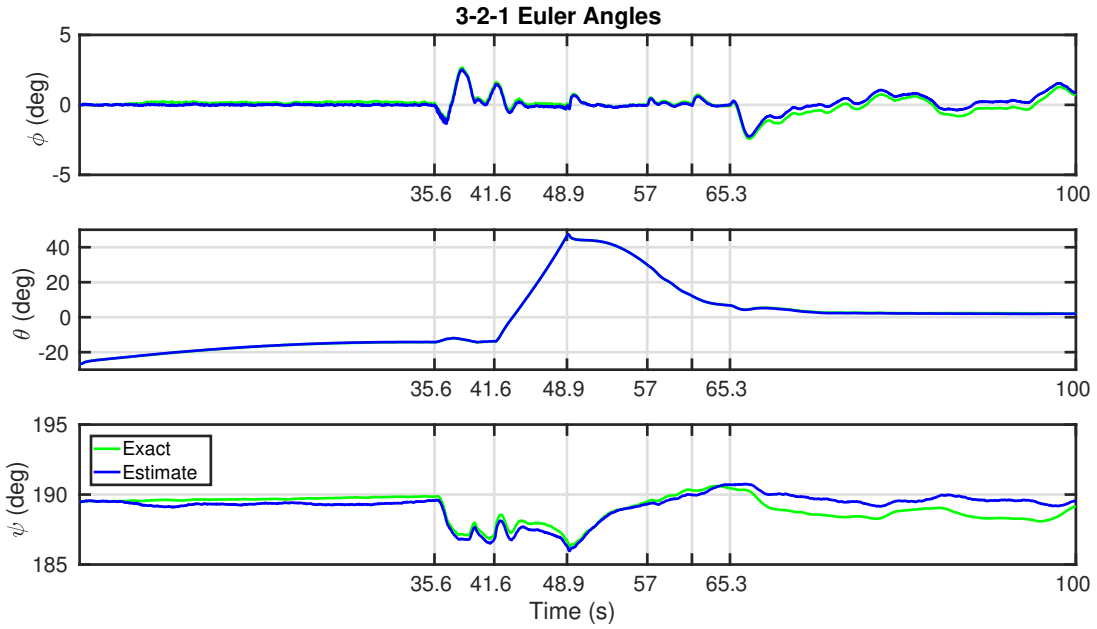


Figure 7.25: Attitude estimate with magnetic field and velocity measurements.

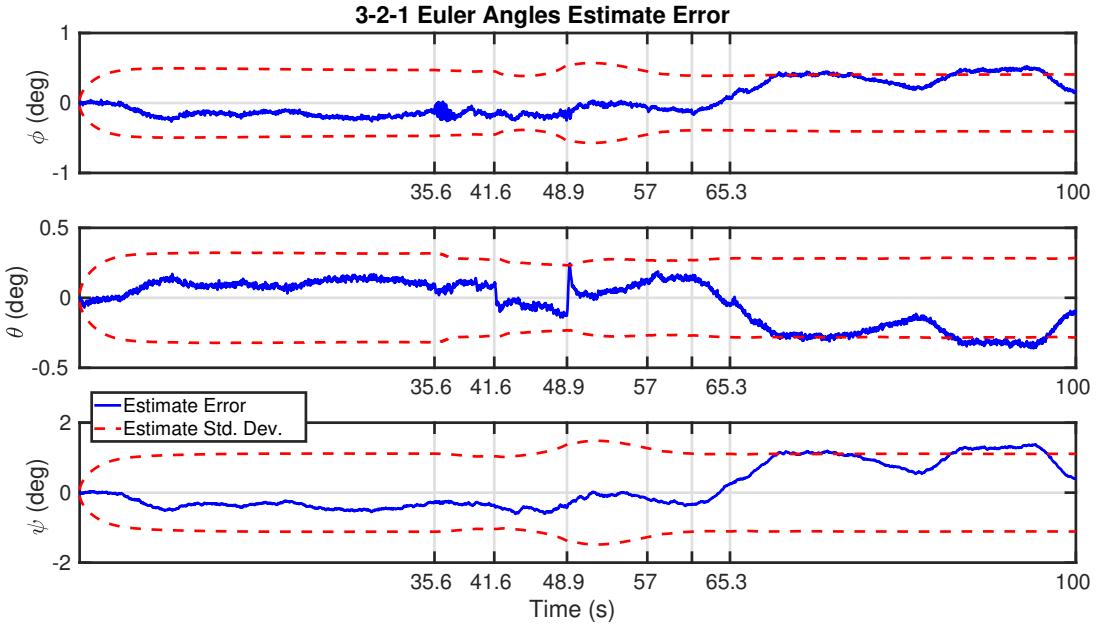


Figure 7.26: Attitude estimate error with magnetic field and velocity measurements.

Figure 7.27 shows the gyroscope bias estimate using the magnetometer and the velocity vector for measurements, and Figure 7.28 shows the estimate error.

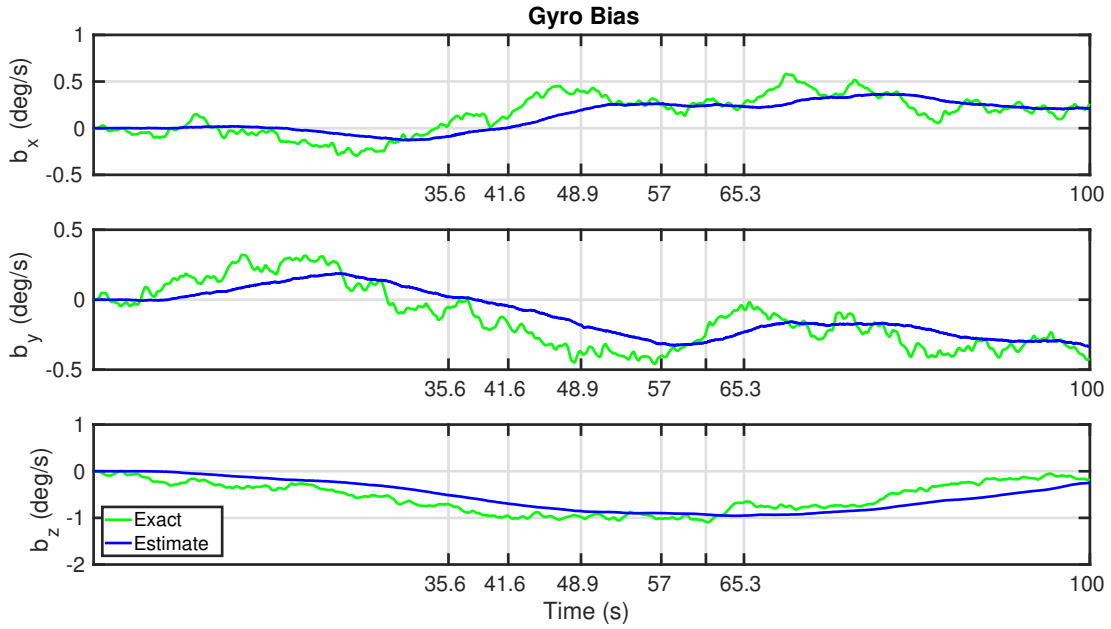


Figure 7.27: Gyroscope bias estimate with magnetic field and velocity measurements.

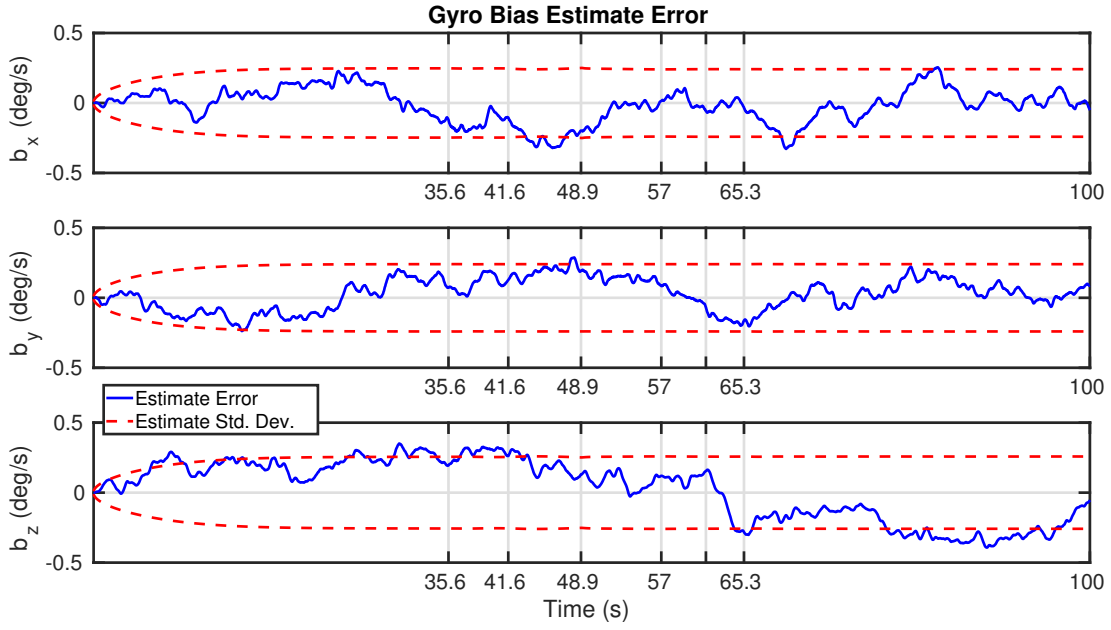


Figure 7.28: Gyroscope bias estimate error with magnetic field and velocity measurements.

The estimation error statistics are given in Table 7.3.

Table 7.3: Estimate errors for magnetometer and velocity vector measurements.

Estimation Error	RMS Error	Standard Deviation	Max Error
Roll Angle (deg)	0.2470	0.2435	0.5248
Pitch Angle (deg)	0.1670	0.1628	0.3614
Yaw Angle (deg)	0.6574	0.6423	1.377
Bias x-axis (deg/s)	0.1238	0.1180	0.3286
Bias y-axis (deg/s)	0.1225	0.1149	0.2882
Bias z-axis (deg/s)	0.3286	0.2165	0.3924

Using the velocity vector and the magnetic field for measurements also significantly improves the estimate as compared to the magnetic field measurements alone. However, the results presented thus far were obtained when only wind gusts are simulated. Since the measurement of the velocity vector in body coordinates will also include the velocity of the air, a static wind will cause the inertial velocity vector and the body velocity vector measurements to differ. Figure 7.29 shows the attitude estimate using the magnetometer and the velocity vector for measurements with static winds from the HWM present. Figure 7.30 shows the attitude estimate error.

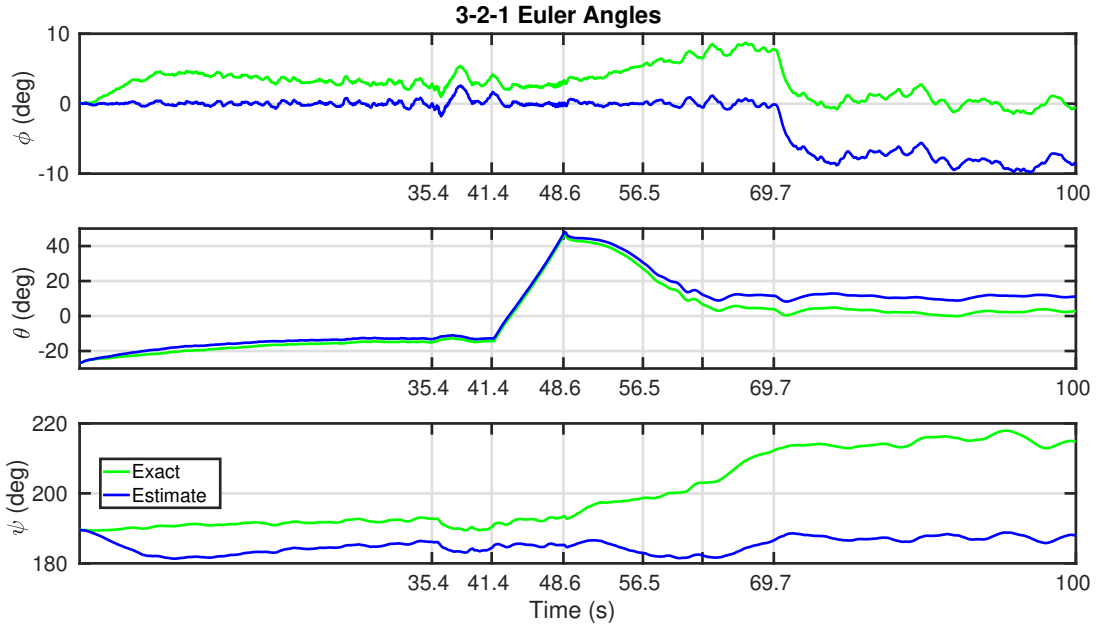


Figure 7.29: Attitude estimate with magnetic field and velocity measurements with static winds.

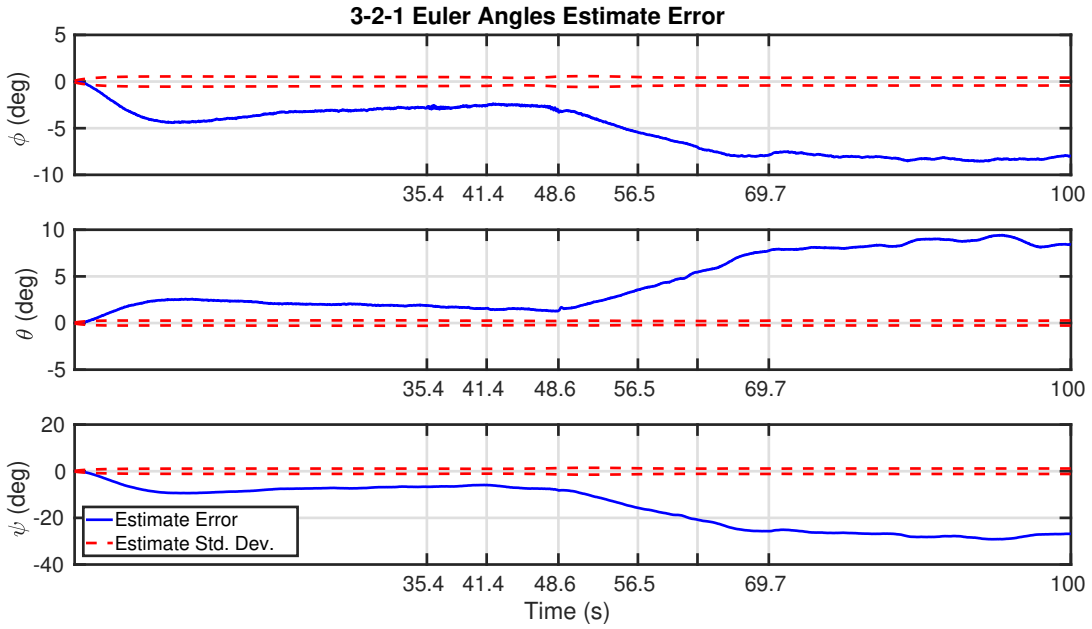


Figure 7.30: Attitude estimate error with magnetic field and velocity measurements with static winds.

Figure 7.31 shows the gyroscope bias estimate using the magnetometer and the velocity vector for measurements with static winds present, and Figure 7.32 shows the estimate error.

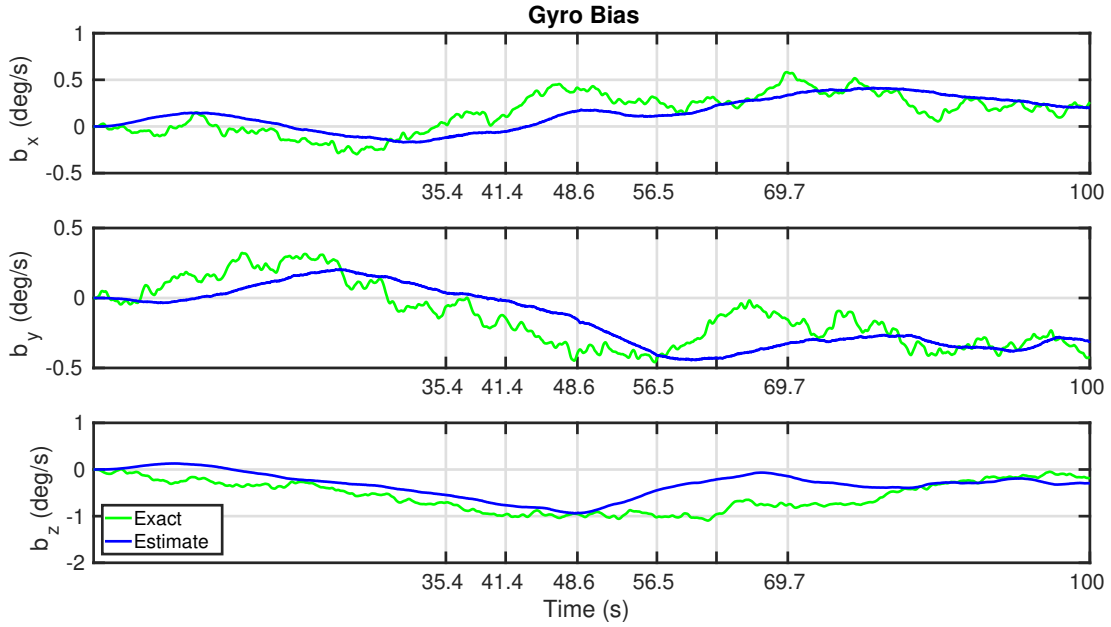


Figure 7.31: Gyroscope bias estimate with magnetic field and velocity measurements with static winds.

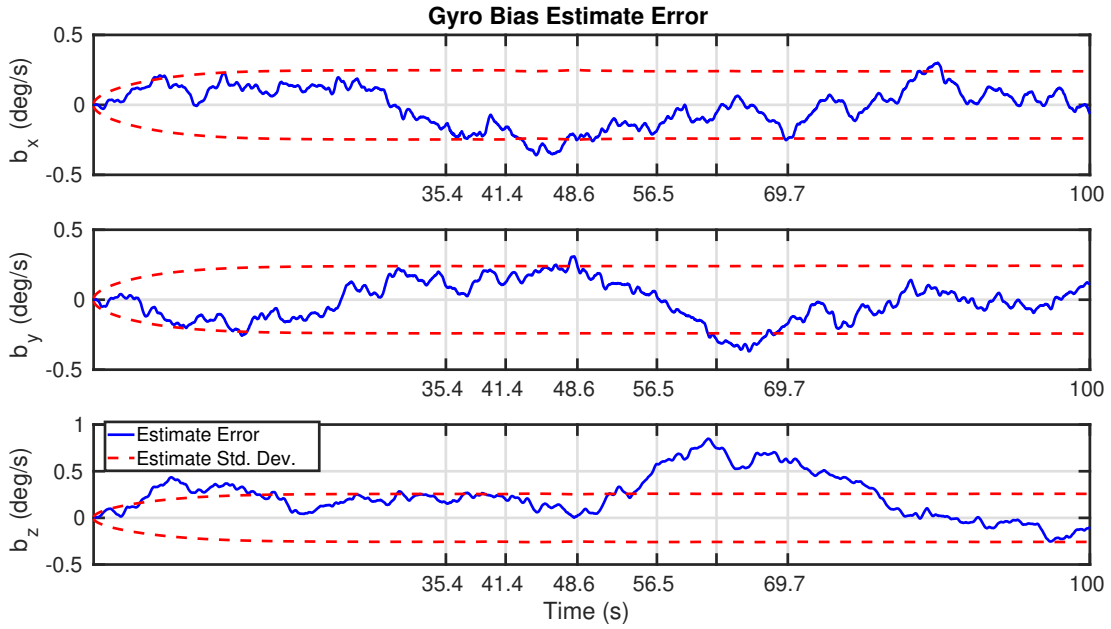


Figure 7.32: Gyroscope bias estimate error with magnetic field and velocity measurements with static wind.

The estimation error statistics are given in Table 7.4.

Table 7.4: Estimate errors for magnetometer and velocity vector measurements with static winds.

Estimation Error	RMS Error	Standard Deviation	Max Error
Roll Angle (deg)	5.743	2.445	8.553
Pitch Angle (deg)	5.347	3.041	9.343
Yaw Angle (deg)	17.96	9.310	29.16
Bias x-axis (deg/s)	0.1441	0.1437	0.3599
Bias y-axis (deg/s)	0.1442	0.1430	0.3688
Bias z-axis (deg/s)	0.3339	0.2356	0.8489

The primary component of the static winds is in the east direction, while the booster is simulated as flying south. This results in a significant component of velocity that differs between the airspeed vector measurement and the inertial velocity measurement. Note that if the booster were flying east or west, this estimate would improve significantly as the reference vectors are normalised before use in the attitude MEKF. It is however unreasonable to assume the direction of the wind will be known, and therefore the velocity vector is not feasible for use as a reference vector in the MEKF without a direct measurement of atmospheric winds. Therefore, Earth's magnetic and gravity fields are chosen as the inertial reference vectors for use in the attitude MEKF.

7.3.2 Position and Velocity Estimate

The inertial position and velocity estimates presented below use the attitude estimate with magnetic field and gravity vector measurements, and also simulate the static winds including turbulence. This section

considers the method selected to handle the measurement delay on the GPS, which is the use of the state estimate corresponding to the time step of the measurement delay in the innovation term for the current time step. Figure 7.33 shows the position estimate and Figure 7.34 shows the position estimate error for the wing deployment when the current position and velocity state estimates are used in the innovation.

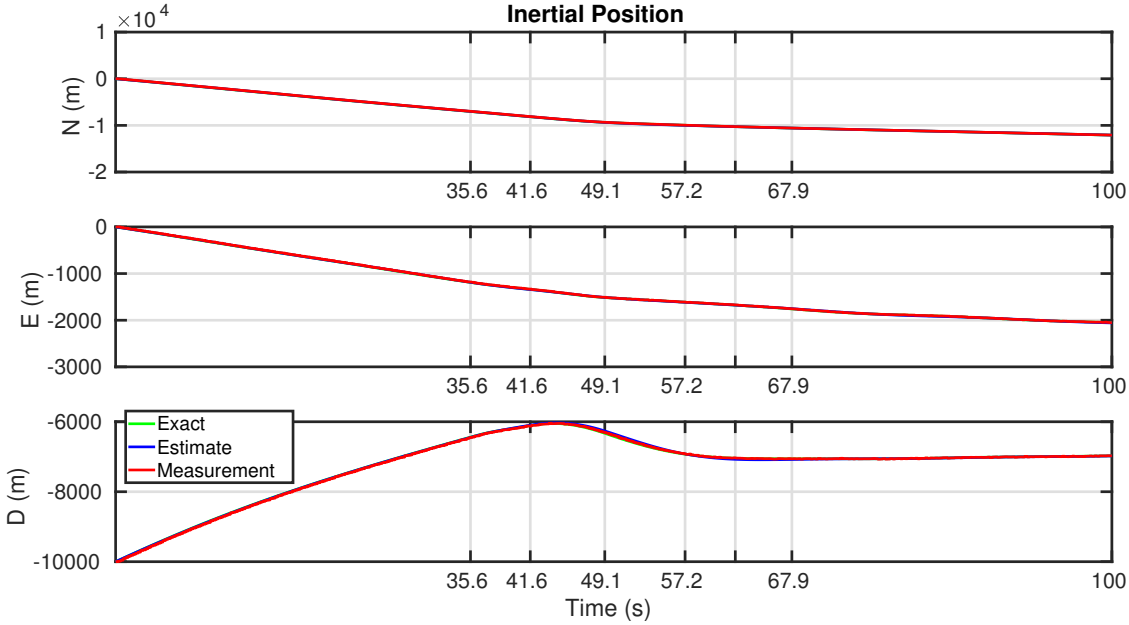


Figure 7.33: Inertial position estimate using the current state in the innovation.

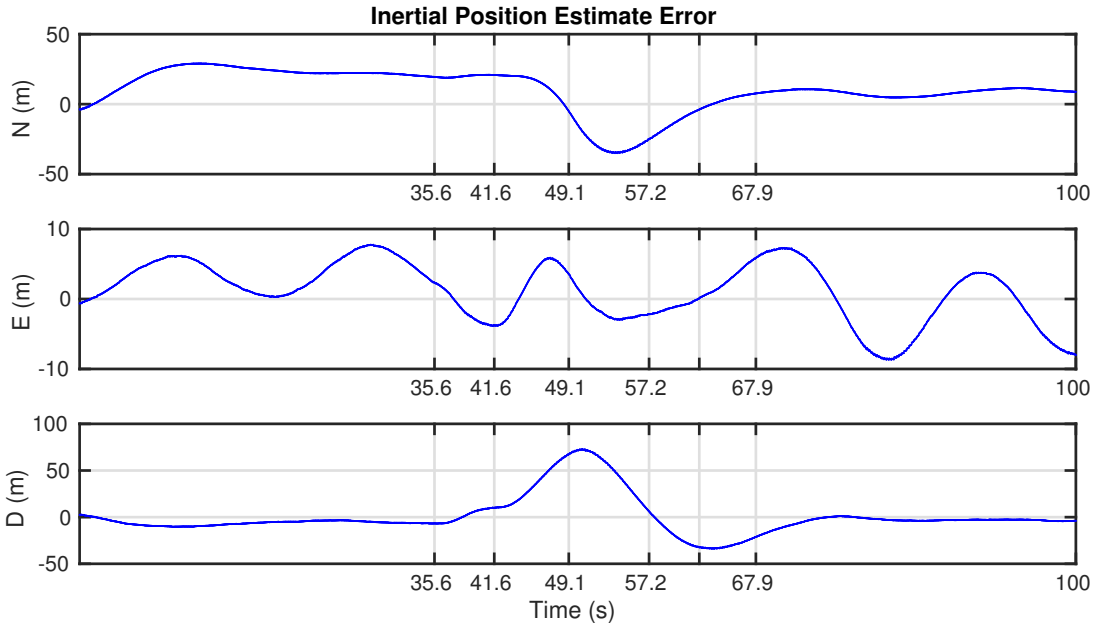


Figure 7.34: Inertial position estimate error using the current state in the innovation.

Figure 7.35 shows the velocity estimate and Figure 7.36 shows the velocity estimate error for the wing deployment when the current position and velocity state estimates are used in the innovation.

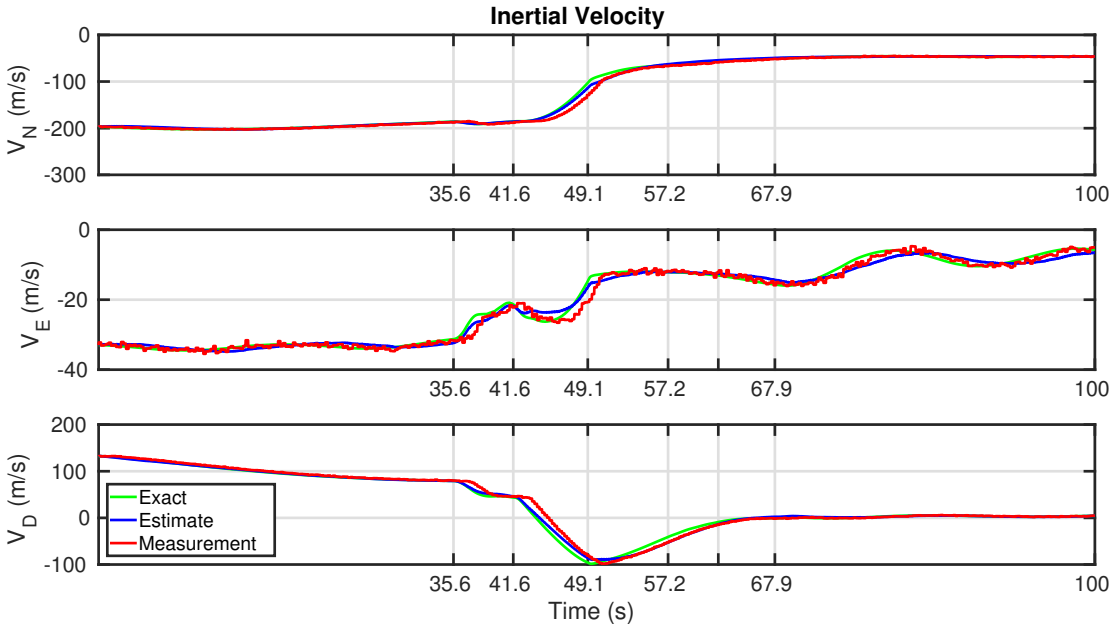


Figure 7.35: Inertial velocity estimate using the current state in the innovation.

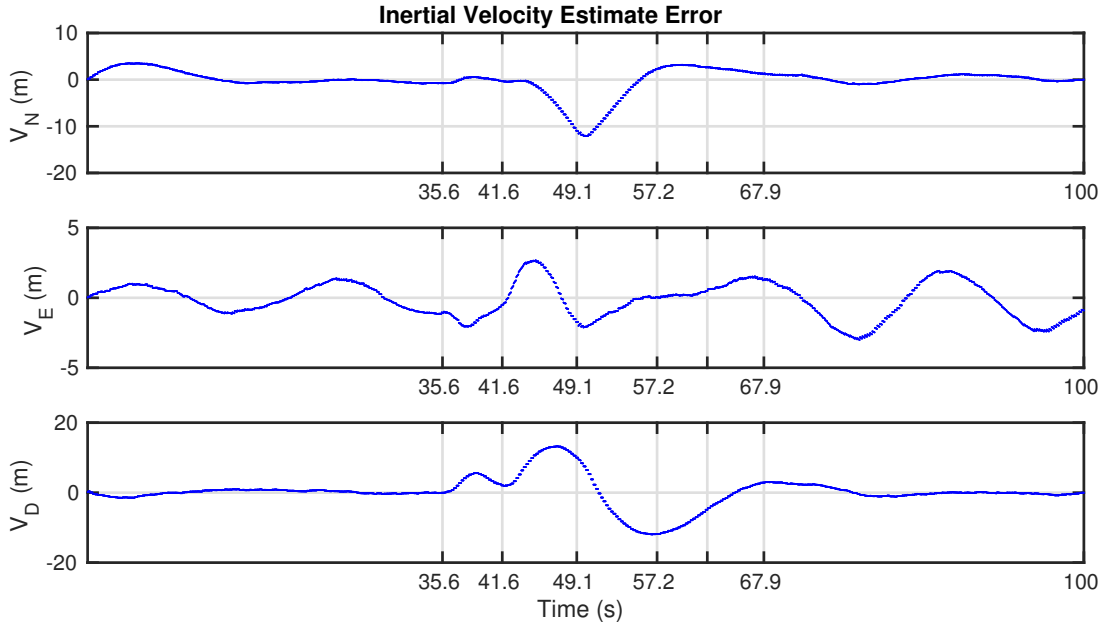


Figure 7.36: Inertial velocity estimate error using the current state in the innovation.

The estimate error statistics are shown in Table 7.5.

Table 7.5: Estimate errors for inertial position and velocity.

Estimation Error	RMS Error	Standard Deviation	Max Error
N (m)	18.01	15.10	35.05
E (m)	4.311	4.162	8.684
D (m)	21.16	21.15	73.10
V_N (m/s)	2.716	2.712	12.09
V_E (m/s)	1.264	1.253	3.017
V_D (m/s)	4.404	4.401	13.25

The 1.1 s measurement delay on the GPS can be clearly seen during the manoeuvre in Figure **??**. When the current state is used in the innovation, the delay on the GPS measurements causes significant errors in the estimate due to a large difference between the current measurement and the current state. The northern and downward states have the most error. This is an expected result since the booster is travelling south and with non-zero flight path angles, so these directions will be significantly affected by the GPS measurement delay. Figure 7.37 shows the position estimate and Figure 7.38 shows the position estimate error for the wing deployment, where the delayed state has been used in the innovation term.

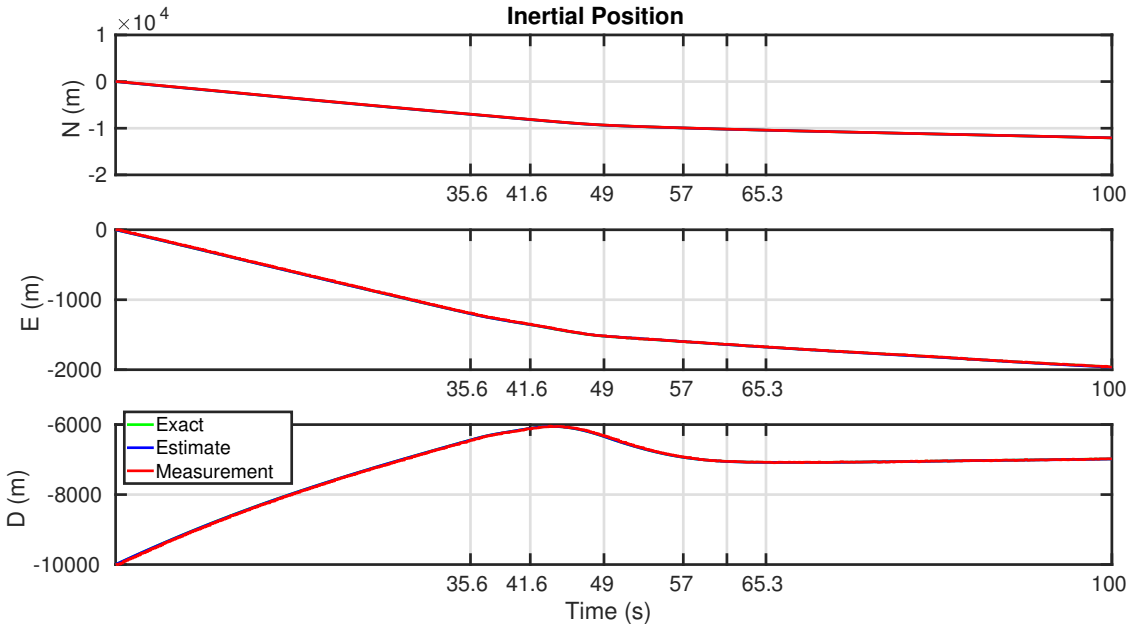


Figure 7.37: Inertial position estimate using the delayed state in the innovation.

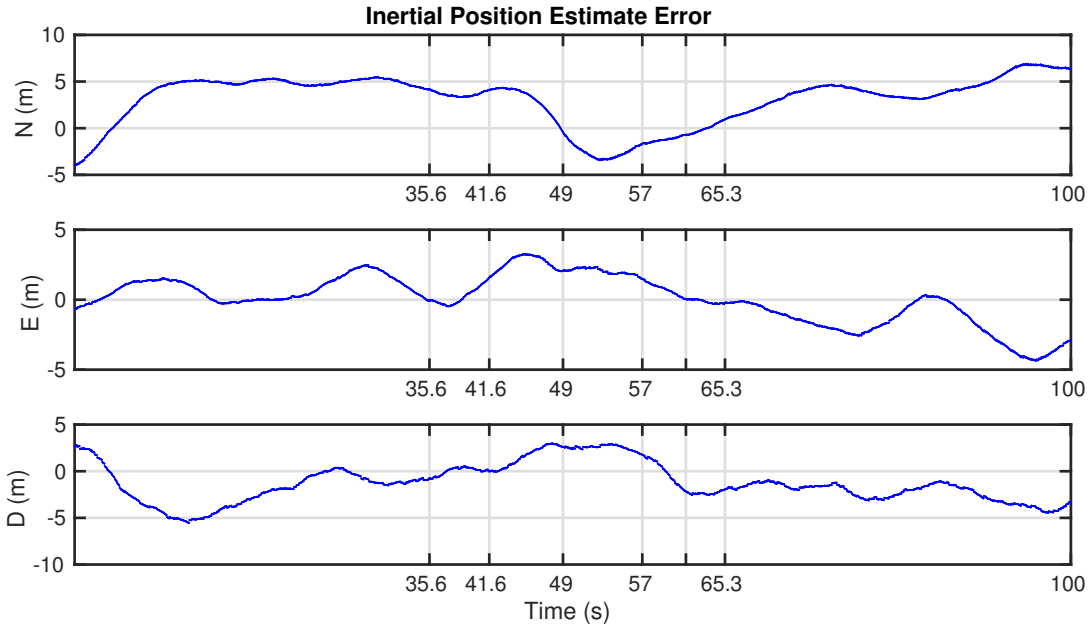


Figure 7.38: Inertial position estimate error using the delayed state in the innovation.

Figure 7.39 shows the velocity estimate and Figure 7.40 shows the velocity estimate error for the wing deployment using the delayed state in the innovation.

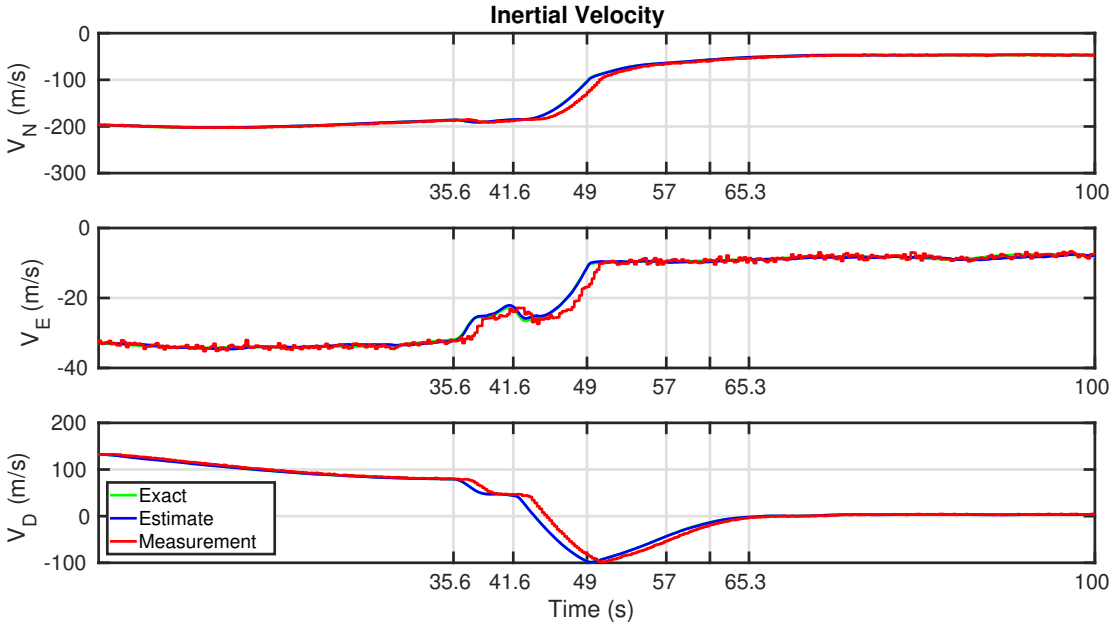


Figure 7.39: Inertial velocity estimate using the delayed state in the innovation.

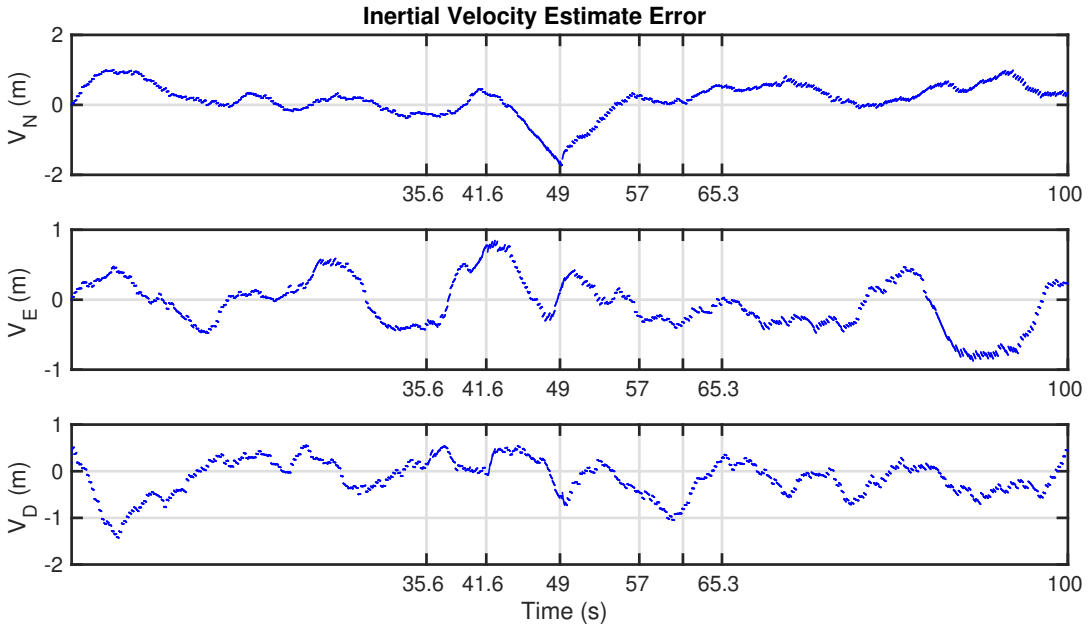


Figure 7.40: Inertial velocity estimate error using the delayed state in the innovation.

The estimate error statistics are shown in Table 7.6.

Table 7.6: Estimate errors for inertial position and velocity using the delayed state in the innovation.

Estimation Error	RMS Error	Standard Deviation	Max Error
N (m)	4.038	2.689	6.913
E (m)	1.731	1.730	4.371
D (m)	2.458	2.116	5.539
V_N (m/s)	0.4997	0.4714	1.731
V_E (m/s)	0.3626	0.3598	0.8749
V_D (m/s)	0.4134	0.3883	1.431

Using the delayed state in the innovation results in significantly better performance, however the north and down positions are still the least accurate. This shows that the using the delayed state in the innovation term does not completely compensate for these delayed measurements, and future work should consider modelling the delayed states using a Padé approximation, as described in [25].

7.3.3 Wind Velocity Estimate

All of the simulation results presented thus far assume that sensors are available to provide a direct measurement of the angle of attack and the sideslip angle. Although a Flush Air Data Sensing (FADS) system is the recommended sensor to provide these measurements, the development and design of a FADS system is not within the scope of this work. Instead we considered a wind velocity estimator that uses a measurement from a pitot-static probe, along with the attitude and inertial velocity estimates, to estimate the wind velocity vector. From this the estimated wind velocity vector and the booster's inertial velocity estimate, the airspeed, angle of attack and sideslip angle may be computed. Figure 7.41 shows the estimation results

for an initial heading of 180°, where it has been assumed that the estimator's initial condition is accurate. Figure 7.42 shows the resulting polar velocity estimate. Since the largest component of the wind velocity is in the east direction, the booster is unable to sense this component while flying south. The estimated east component of wind stays constant at the initial condition until the booster deploys the wing and performs the pull-up manoeuvre. This exemplifies the observability requirement that the booster pitch and yaw in order to observe all three components of wind with the one measurement along the body axis.

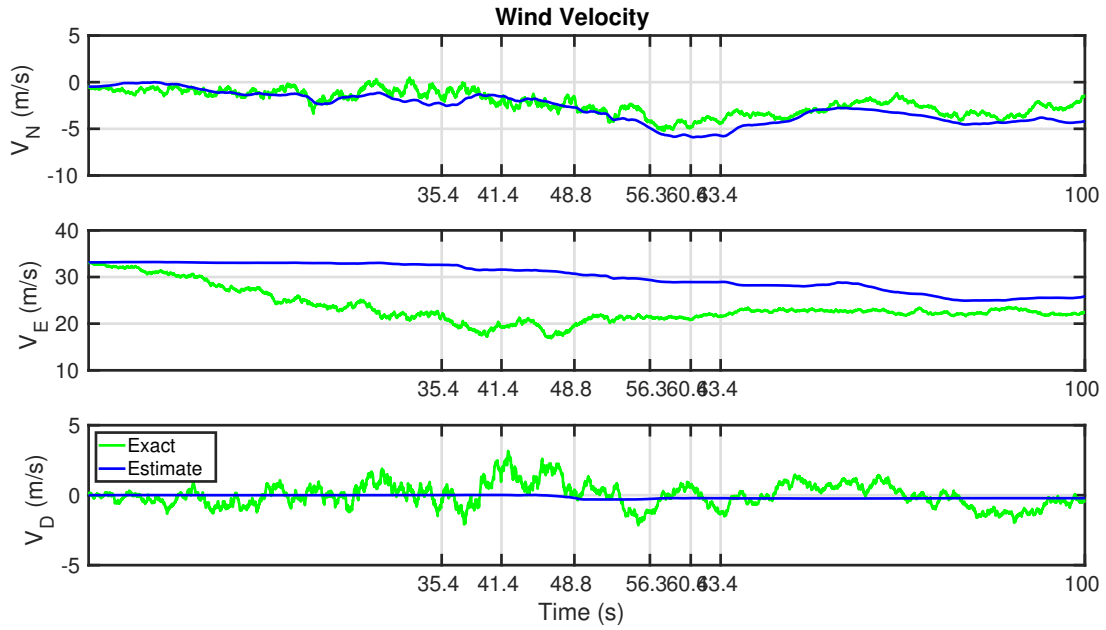


Figure 7.41: Wind velocity estimate for initial heading of 180°.

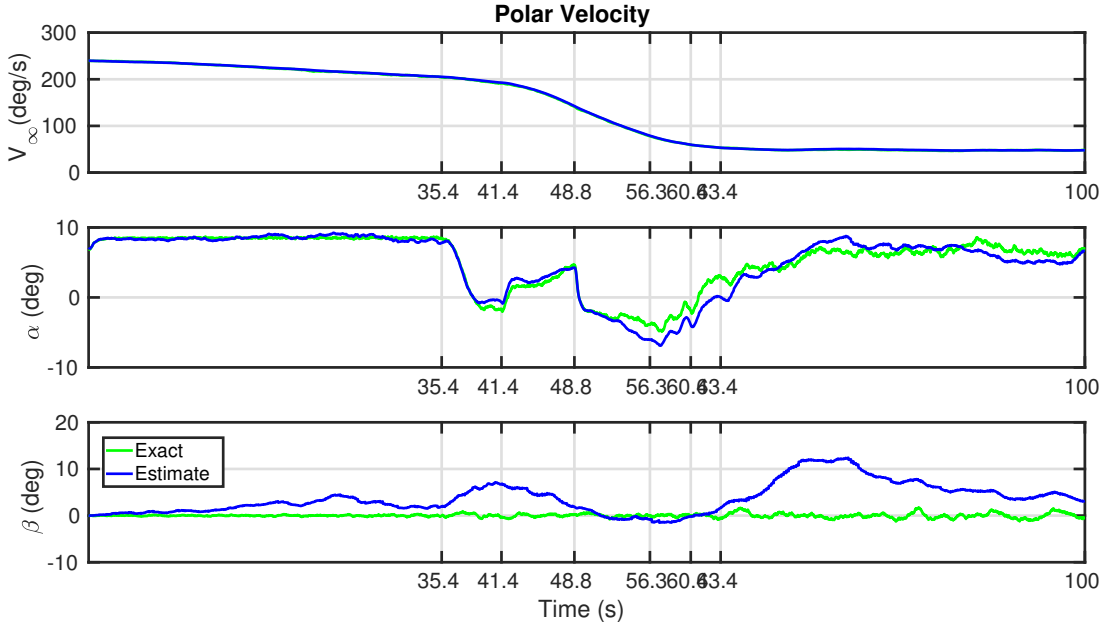


Figure 7.42: Polar velocity estimate for initial heading of 180°.

Figure 7.43 shows the wind velocity estimate for an initial heading of 270°, and Figure 7.44 shows the

resulting polar velocity estimate. In this case, the estimator is able to observe the major component of the wind velocity and the polar velocity estimate is good enough to use in the control system.

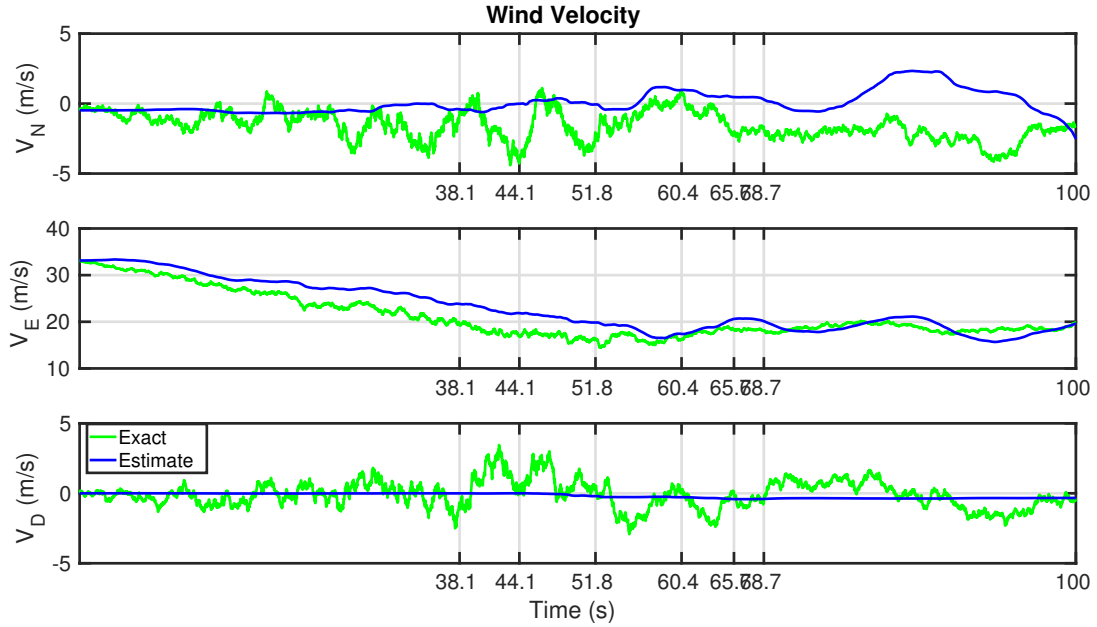


Figure 7.43: Wind velocity estimate for initial heading of 270°.

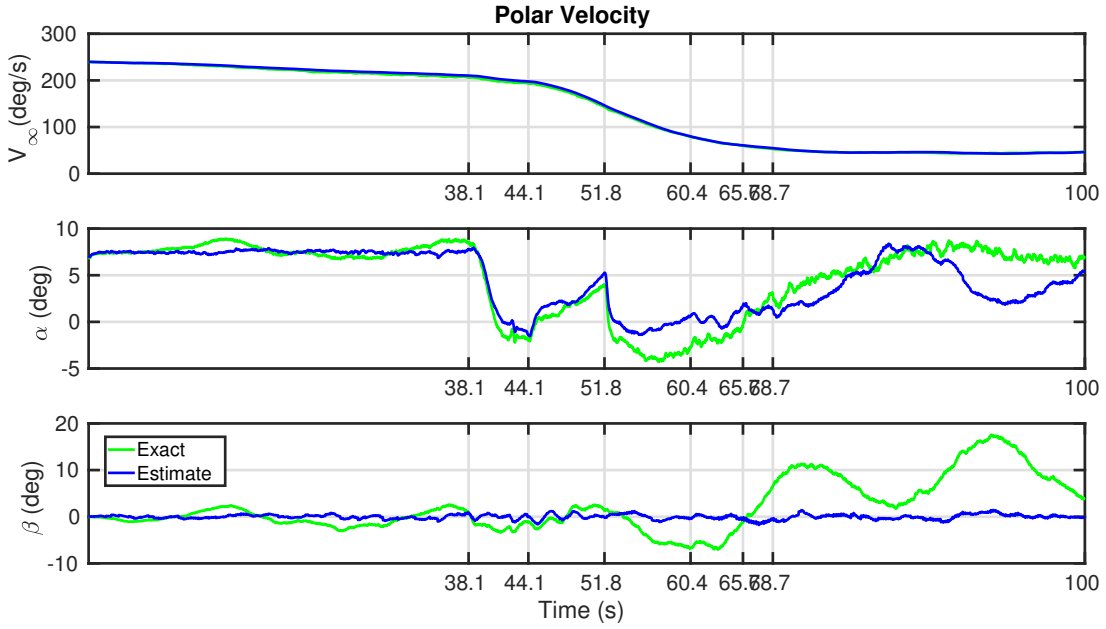


Figure 7.44: Polar velocity estimate for initial heading of 270°.

In [62], the observability of a wind velocity estimator using the FADS on the Mars Science Laboratory was shown to increase during the bank reversals which occurred during atmospheric entry. Performing bank reversals was also used by the Space Shuttle to decrease velocity throughout reentry into Earth's atmosphere. Thus, a series of bank reversals were implemented in the rocket configuration to investigate the effect of this manoeuvre on the wind velocity estimator implemented in this work. Figure 7.45 shows this

manoeuvre, in which two 30° step inputs are provided to the attitude controller, and rate limited at 5° per second to prevent overshoot. This bank reversal achieves a change in heading of 20°. The resulting wind velocity estimate and calculated polar velocity are shown in Figure 7.47. There is a very small improvement in the estimation of the east component of the wind velocity due to the bank reversal as compared to Figure 7.42. However, the bank reversal investigated here does not provide enough additional information of the wind velocity in order to achieve a sufficient calculation of the sideslip angle.

One reason for this is that the change in heading resulting from this bank reversal is not very significant. Banking at a larger roll angle will provide a larger change in heading, however it should be noted that the ALV first stage boosters do not produce a significant amount of lift in the rocket configuration. Although a FADS inherently provides more information about the local flow angles, future work should design the FADS without relying on atmospheric wind estimations aided by bank reversals.

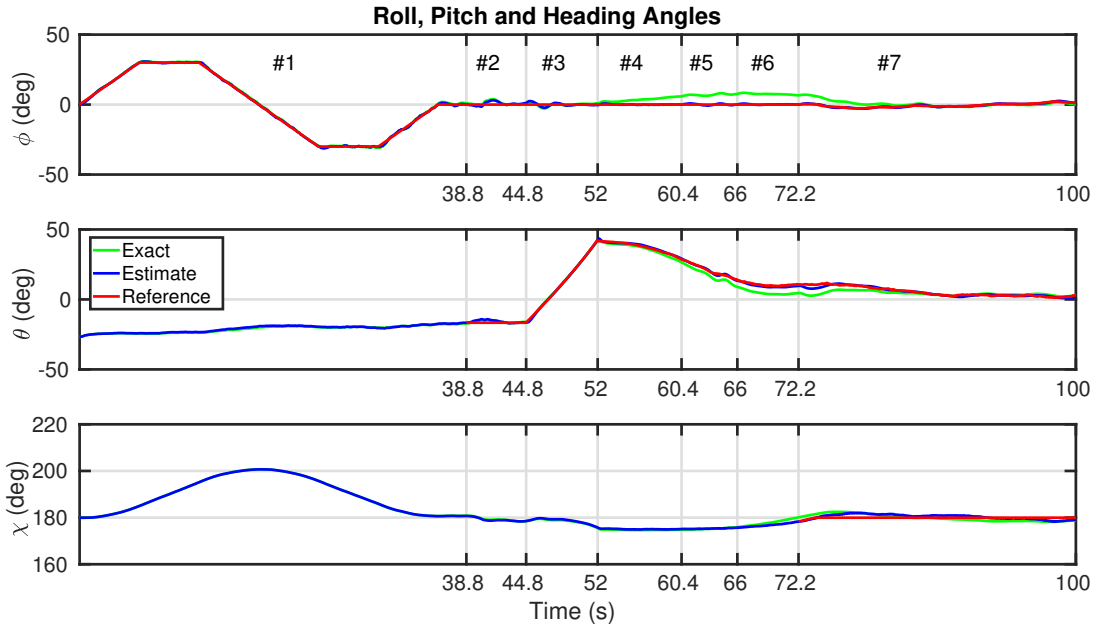


Figure 7.45: Roll angle, pitch angle and heading for a series of bank reversals in the rocket configuration.

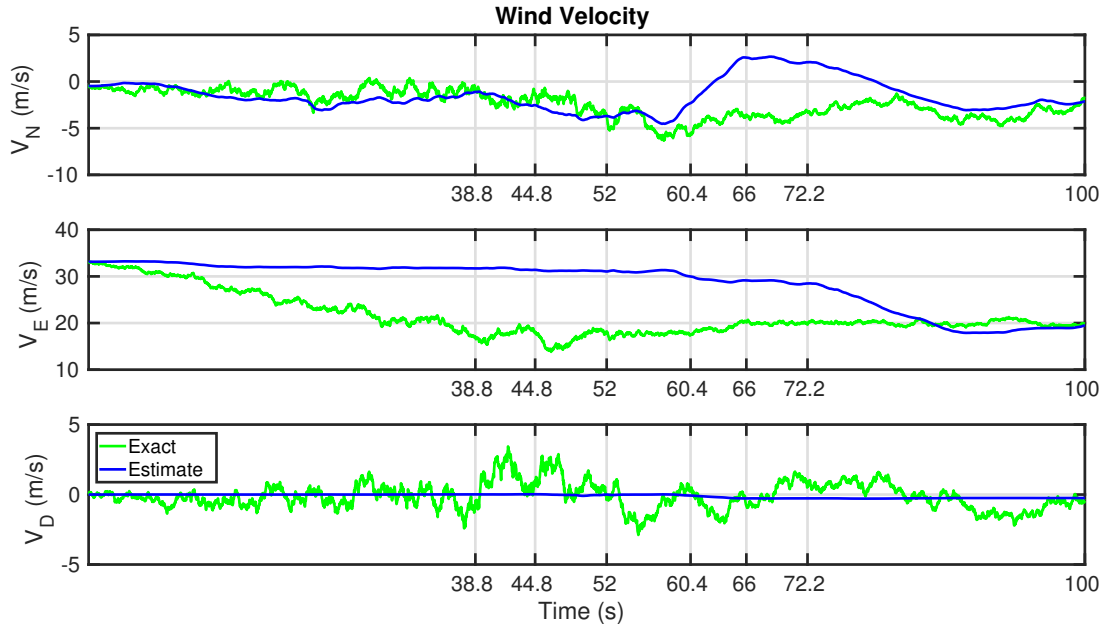


Figure 7.46: Wind velocity estimate for initial heading of 270° with a series of bank reversals.

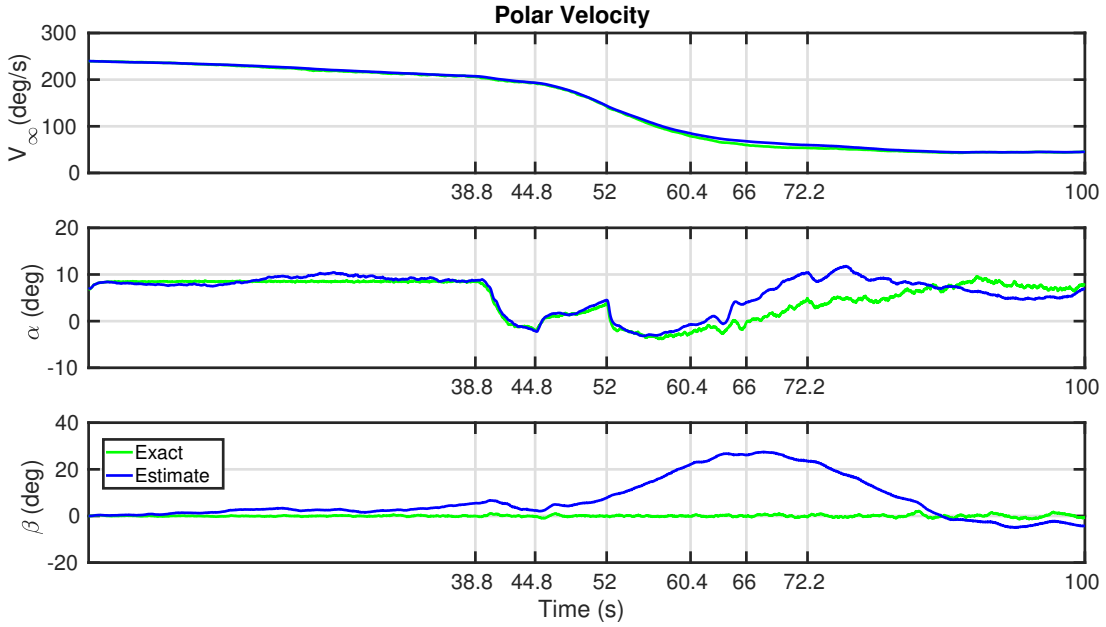


Figure 7.47: Polar velocity estimate for initial heading of 270° with a series of bank reversals.

7.4 Conclusion

This chapter has compared the various control and navigation designs described in Chapters 5 and 6, respectively. The control system simulation results showed that control allocation is useful for dealing with control surface saturation expected during the initial wing deployment, however may lead to situations where the control effectors are not ideally utilised due to the control allocation problem being underdetermined. This was observed in the aircraft configuration, where ailerons were used to contribute to a

pitching moment. This work reverted to explicit ganging once the wing was fully deploy to circumvent this issue. Furthermore, the approximation that the angle of attack does not vary significantly during the pull-up manoeuvre was shown to be reasonable. Therefore, the airspeed controller used to achieve the minimum sink rate glide was activated as soon as the wing partially deploys to a sweep angle of 60°, and the commanded pitch angle was rate limited to remain within the booster's load factor limits during the pull-up manoeuvre. Since the airspeed error will initially be very large, a nonlinear proportional controller was implemented. The proportional term is the hypebolic tangent function, which saturates at large airspeed errors while reducing to a linear controller at small airspeed errors. Finally, the heading angle controller was shown to be ineffective prior to obtaining the minimum sink rate glide, and is therefore only activated after fully deploying the wing.

The navigation system simulation results showed that Earth's magnetic and gravitational fields are the most reliable to use as inertial reference vectors in the attitude MEKF. Using the booster's velocity vector is unfeasible as atmospheric winds causes the velocity measured by the air data computer to vary significantly from the velocity measured by the inertial position and velocity estimator. For the inertial position and velocity estimator, the delay on the GPS measurement was shown to cause significant errors if it is not considered in the Kalman filter. The computationally simple method of using the delayed state in the filter innovation to compensate for the measurement delay was shown to significantly improve the estimate. Finally, a wind velocity observer using a measurement from a pitot-static probe and the outputs from the attitude MEKF and inertial velocity estimator to construct the flow angles was considered. The estimate of the angle of attack and sideslip angles was severely degraded when the major component of the wind velocity is not aligned with the pitot-static probe. Furthermore, performing bank reversals, which achieved a change in heading of 20° did not add significant additional information to the estimate. The recommended method for obtaining the flow angles is therefore a Flush Air Data Sensing (FADS) system.

Chapter 8

Summary and Recommendations

8.1 Summary

This research has defined a strategy for deploying the wing of the Austral Launch Vehicle's first stage booster prior to initiating the flyback portion of the return flight. First, an aerodynamic analysis modelling capability has been developed using a potential flow panel method. The critical Mach number of the booster's wing was investigated with this potential flow panel method, as well as with computational fluid dynamic analysis. The results showed that the critical Mach number of the partially deployed wing is significantly higher than the fully deployed wing. Thus, the wing is first partially deployed, and then a pull-up manoeuvre is initiated in order to further decrease velocity and to regain altitude. Once the wing is fully deployed, a minimum sink rate glide is then obtained to maximise the time available to start the piston engine for the flyback portion of the return flight.

A flight control system has been developed to manage the transition between the rocket configuration and the aircraft configuration. A control allocation algorithm manages the varying availability of the aerodynamic control surfaces. A cascaded control architecture then manages the highly coupled dynamics resulting from the partial deployment of the wing. This control architecture consists of an angular velocity inner loop and an attitude outer loop developed using linear quadratic servo control theory. To achieve the flight objectives, outer loop single input, single output controllers were developed to perform the pull-up manoeuvre and then to achieve a minimum sink rate glide.

8.2 Contributions

The original contributions made in this work are:

- Established an aerodynamic modelling capability to simulate the atypical dynamics of the oblique wing configuration using NASA's OpenVSP to model the geometry of the booster, and Warsaw Technical University's PANUKL to obtain the aerodynamic coefficients. Custom software was developed

to integrate these two software packages and to compute the stability and control derivatives.

- Addition of the elevons to the booster's empennage to provide better lateral-directional disturbance rejection in the rocket configuration.

- Defined the objectives and constraints for a manoeuvre to deploy the wing as early as possible in the return flight and then obtain gliding flight that maximises the time available to start the piston enginer. The wing is first partially deployed to a 60° sweep angle, which has a significantly higher critical Mach number than the fully deployed wing. The booster then pulls-up to exchange kinetic energy for gravitational potential energy, while remaining within the booster's load factor limits, and then attains a minimum sink rate glide with the wing fully deployed.

- Development of a cascaded control architecture that handles the varying booster configuration and control objectives throughout the wing deployment manoeuvre. A control allocation algorithm was designed specifically to handle saturation of the control surfaces during the initial wing deployment. Two cascade multiple-input, multiple-output controllers manage the dynamic cross-coupling resulting from the partial deployment of the wing. Outer-loop controllers are then designed to meet the flight objectives defined by the wing deployment manoeuvre.

- Development of estimators using multiple Kalman filters that handles the large dynamic accelerations resulting from the pull-up manoeuvre, as well as the significant delay on the GPS measurement.

8.3 Future Work

The following items should be addressed in future work:

- The aerodynamic modelling software is not able to accurately model the aerodynamics during the initial wing deployment. Computational fluid dynamics analysis should be performed to accurately model the wing-fuselage interactions during this initial deployment.

- The control allocation algorithm does not prioritise which moment to achieve in the event that the pseudo-inverse becomes overdetermined. The initial wing deployment is expected to result in a highly unstable roll mode, and therefore the control allocation algorithm should be expanded to prioritise maintaining the roll angle likely resulting from the aforementioned un-modelled instability at the high sweep angles.

- The gain scheduling for the transition between the pull-up manoeuvre and the minimum sink rate glide was iteratively designed. There was a trade-off between the amount that the booster pulled up and the amount of altitude regained. Trajectory optimisation could be utilised to achieve a pull-up

manoeuvre and transition to the minimum sink rate glide while remaining within the angle of attack limits of the booster.

- The initial conditions for all simulations were assumed to be at an equilibrium condition at 10 km altitude and Mach 0.8. The hypersonic and supersonic portions of the return flight have not yet been designed, and future work is required to verify the feasibility of the initial conditions assumed in this work.

- The critical Mach number for the partially deployed wing was found to be heavily impacted by a local region of sonic flow near the wing tips at the transition between the straight and tapered portions of the wing. It may be possible to increase the critical Mach number by optimising the shape of the wing tips to prevent this sonic region from forming. This could potentially allow the wing to be partially deployed at a much higher Mach number, expanding the envelope of the oblique wing configuration.

- A test flight of the first stage booster should be performed to verify the designs of the flight control system and navigation system, in particular during the oblique wing configuration and during the initial wind deployment.

References

[1] A. Schutte and P. Thoreau. “The Austral Launch Vehicle: 2014 Progress in Reducing Space Transportation Cost through Reusability, Modularity and Simplicity”. In: *Proceedings of the 12th Reinventing Space Conference*. 2014.

[2] Jr. T. J. Healy. *Shuttle Liquid Fly Back Booster Configuration Options*. 1998.

[3] S. C. Smith, R. A. Kennelly, and J. Reuther. *Oblique-Wing Glide-Back Booster for Shuttle Reusable First Stage*. 1999.

[4] M. J. Hirschberg, D. M. Hart, and T. J. Beutner. “A Summary Of A Half-Century of Oblique Wing Research”. In: *45th AIAA Aerospace Sciences Meeting and Exhibit*. 2007.

[5] Desktop Aeronautics Inc. *Oblique Flying Wings: An Introduction and White Paper*. June 2005.

[6] Alex G. Sim and Robert E. Curry. “Flight Characteristics of the AD-1 Oblique-Wing Research Aircraft”. In: *NASA Technical Reports* 2223 (March 1985).

[7] John H. Blakelock. *Automatic Control of Aircraft and Missiles*. Wiley Interscience, 1991.

[8] Randal W. Beard and Timothy W. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012.

[9] D. F. Enss, D. J. Bugajski, and M. J. Klepl. “Flight Control for the F-8 Oblique Wing Research Aircraft”. In: *American Control Conference*. 1987.

[10] Robert W. Kempel, Walter E. McNeill, Glenn B. Gilyard, et al. “A Piloted Evaluation of an Oblique-Wing Research Aircraft Motion Simulation With Decoupling Control Laws”. In: *NASA Technical Paper 2874* (1988).

[11] Gurbux S. Alag, Robert W. Kempel, and Joseph W. Pahle. “Decoupling Control Synthesis for an Oblique-Wing Aircraft”. In: *NASA Technical Memorandum 86801* (1986).

[12] G. S. Alag, R. W. Kempel, J. W. Pahle, et al. “Model-Following Control for an Oblique-Wing Aircraft”. In: *NASA Technical Memorandum 88269* (1986).

[13] Joseph W. Pahle. “Output Model-Following Control Synthesis for an Oblique-Wing Aircraft”. In: *NASA Technical Memorandum 100454* (1990).

[14] Stephen James Morris. “Integrated Aerodynamic and Control System Design of Oblique Wing Aircraft”. PhD thesis. Stanford University, 1990.

[15] Robert E. Curry and Alex G. Sim. “In-Flight Total Forces Moments and Static Aeroelastic Characteristics of an Oblique-Wing Research Airplane”. In: *NASA Technical Paper 2224* (1984).

[16] Uy-Loi Ly. “A Design Algorithm for Robust Low-Order Controllers”. PhD thesis. Stanford University, Nov. 1982.

[17] Jie Pang, Rong Mei, and Mou Chen. “Modeling and Control for Near-Space Vehicles With Oblique Wing”. In: *Proceedings of the 10th World Congress on Intelligent Control and Automation*. July 2012.

[18] Wang Lixin, Xu Zijian, and Yue Ting. “Dynamic characteristics analysis and flight control design for oblique wing aircraft”. In: *Chinese Journal of Aeronautics* (2016).

[19] Tor A. Johansen and Thor I. Fossen. “Control allocation - A survey”. In: *Automatica* (2013).

[20] R. E. Kalman. “A New Approach to Linear Filtering and Prediction Problems”. In: *Journal of Basic Engineering* (1960).

[21] Andrew Mark Eldredge. “Improved State Estimation for Miniature Air Vehicles”. MA thesis. Brigham Young University, Aug. 2006.

[22] Dongwon Jung and Panagiotis Tsiotras. “Inertial Attitude and Position Reference System Development for a Small UAV”. In: *AIAA Infotech@Aerospace 2007 Conference and Exhibit*. 2007.

[23] E. J. Lefferts, F. L. Markley, and M. D. Shuster. “Kalman Filtering for Spacecraft Attitude Estimation”. In: *AIAA 20th Aerospace Sciences Meeting*. Jan. 1982.

[24] F. L. Markley. “Attitude Error Representations for Kalman Filtering”. In: *Journal of Guidance, Control and Dynamics* (2003).

[25] Johan Bijker. “Development of an Atittude Heading Reference System for an Airship”. MA thesis. Stellenbosch University, Dec. 2006.

[26] C. Kamali and Shikha Jain. “Multiplicative Error State Kalman Filter vs Nonlinear Complimentary Filter for a High Performance Aircraft Attitude Estimation”. In: *Defense Science Journal* (Nov. 2016).

[27] J. F. Guerrero Castellanos, S. Lesecq, N. Marchand, et al. “A Low-Cost Air Data Attitude Heading Reference System for the Tourism Airplane Applications”. In: *IEEE Sensors*. 2005.

[28] Douglas Weibel, Dale Lawrence, and Scott Palo. “Small Unmanned Aerial System Attitude Estimation for Flight in Wind”. In: *Journal of Guidance, Control, and Dynamics* (2015).

[29] T. D. Larsen, N. A. Andersen, O. Ravn, et al. “Incorporation of Time Delayed Measurements in a Discrete-time Kalman Filter”. In: *37th IEEE Conference on Decision and Control*. 1998.

[30] François Martel, Parimal K. Pal, and Mark L. Psiaki. “Three-Axis Attitude Determination via Kalman Filtering of Magnetometer Data”. In: *Flight Mechanics/Estimation Theory Symposium*. 1989.

[31] James K. Hall, Nathan B. Knoebel, and Timothy W. McLain. “Quaternion Attitude Estimation for Miniature Air Vehicles Using a Multiplicative Extended Kalman Filter”. In: *Position, Location and Navigation Symposium*. 2008.

[32] Youngjoo Kim, Seunghan Lim, and Hyochoong Bang. “Attitude Estimation for Unmanned Aircraft Using a Multiplicative Extended Kalman Filter”. In: *Asia-Pacific International Symposium on Aerospace Technology*. 2013.

[33] T. A. Johansen, A. Cristofaro, and K. Sørenson. “On estimation of wind velocity, angle-of-attack and sidelip angle of small UAVs using standard sensors”. In: *2015 International Conference on Unmanned Aircraft Systems*. June 2015.

[34] III P. M. Siemers. *Shuttle Entry Air Data System*. 1978.

[35] R. P. G. Collinson. *Introduction to Avionics*. Chapman & Hall, 1996.

[36] M. C. Koen. “Modelling and Simulation of an RPV for Flight Control System Design Purposes”. In: *International Congress of the Aeronautical Sciences* (2000).

[37] Jr. John D. Anderson. *Fundamentals of Aerodynamics*. McGraw-Hill, 2007.

[38] M. Niţă and D. Scholz. “Estimating the Oswald Factor From Basic Aircraft Geometrical Parameters”. In: *Deutscher Luft- und Ramfahrtkongress*. 2012.

[39] Frank Hitchens. *The Encyclopedia of Aerodynamics*. Andrews UK Limited, 2015.

[40] Iain Peddle and Japie Engelbrecht. *Advance Automation 833: Introductory course to aircraft dynamics*. April 2016.

[41] *Wings and Configurations for High-Speed Flight*. URL: `https://history.nasa.gov/SP-468/ch10-4.htm`.

[42] Francisco Palacios, Michael R. Colonno, Aniket C. Aranake, et al. “Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design”. In: *51st AIAA Aerospace Sciences Meeting*. 2013.

[43] Christophe Geuzaine and Jean-François Remacle. “Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities”. In: *International Journal for Numerical Methods in Engineering* (2009).

[44] Daniel P. Raymer. *Aircraft design: a conceptual approach*. American Institute of Aeronautics and Astronautics, 1992.

[45] Louis V. Schmidt. *Intoduction to Aircraft Flight Dynamics*. AIAA Education Series. American Institute of Aeronautics and Astronautics, 1998.

[46] Frank L. Lewis and Vassilis L. Syrmos. *Optimal Control*. John Wiley and Sons, Inc., 1995.

[47] Michael Athans. *Viewgraph Presentations on Multivariable Control Systems*. June 1985.

[48] Huibert Kwakernaak and Raphael Sivan. *Linear Optimal Control Systems*. Wiley-Interscience, 1972.

[49] Michael W. Oppenheimer, David B. Doman, and Michael A. Bolender. In: *14th Mediterranean Conference on Control and Automation*. June 2006.

[50] Wayne Durham, Kenneth A. Bordignon, and Roger Beck. *Aircraft Control Allocation*. Wiley & Sons, Ltd, 2017.

[51] Jon C. Harpold and Claude A. Graves. *Shuttle Entry Guidance*. 1979.

[52] Jingqing Han. "From PID to Active Disturbance Rejection Control". In: *IEEE Transactions on Industrial Electronics* 56.3 (Mar. 2009).

[53] Inc Analog Devices. *High Precision Tri-Axis Inertial Sensor*. 2009.

[54] u-blox AG. *Antaris 4 GPS Modules Data Sheet*. 2007.

[55] Stephan A. Whitemore, Brent R. Cobleigh, and Edward A. Haering. *Design and Calibration of the X-33 Flush Airdata Sensing (FADS) System*. 1998.

[56] G. Chen, B. Chen, P. Li, et al. "Study on Algorithms of Flush Air Data Sensing System for Hypersonic Vehicle". In: *Procedia Engineering* (2014).

[57] M. Jayakumar, N. Remesh, A. Kumar, et al. "A Novel Method for Computation of Air Data Parameters using Nose Cap Pressure Distribution". In: *National Conference on Advanced Computing, Communication and Electrical Systems*. 2017.

[58] Dan Simon. *Optimal State Estimation*. John Wiley and Sons, Inc., 2006.

[59] E. Thébault, C. C. Finlay, C. D. Beggan, et al. "International Geomagnetic Reference Field: the 12th generation". In: *Earth, Planets and Space* (2015).

[60] Douglas P. Drob, John T. Emmert, John W. Meriwether, et al. *An update to the Horizontal Wind Model (HWM): The quiet time thermosphere*. July 2015.

[61] Anonymous. *Part 23 - Airworthiness Standards: Normal Category Airplanes*. 2017. URL: `https://www.ecfr.gov/cgi-bin/text-idx?SID=685dc1ae97ae3f5e5569e47880fab01e&mc=true&node=pt14.1.23`.

[62] Christopher D. Karlgaard, Prasad Kutty, and Mark Schoenenberger. "Couple Inertial Navigation and Flush Air Data Sensing Algorithm for Atmosphere Estimation". In: *Journal of Spacecraft and Rockets* 54.1 (2017).

Appendix A

First Stage Booster Geometry

Figure A.1 shows a top view of the booster geometry. The fuselage has a length l and diameter d. The wing has a span at zero sweep of b_0, a chord length of c_0, a planform area of S and a aspect ratio of $\mathcal{AR}$. The tail has a span of b_t and a chord length of c_t. Table

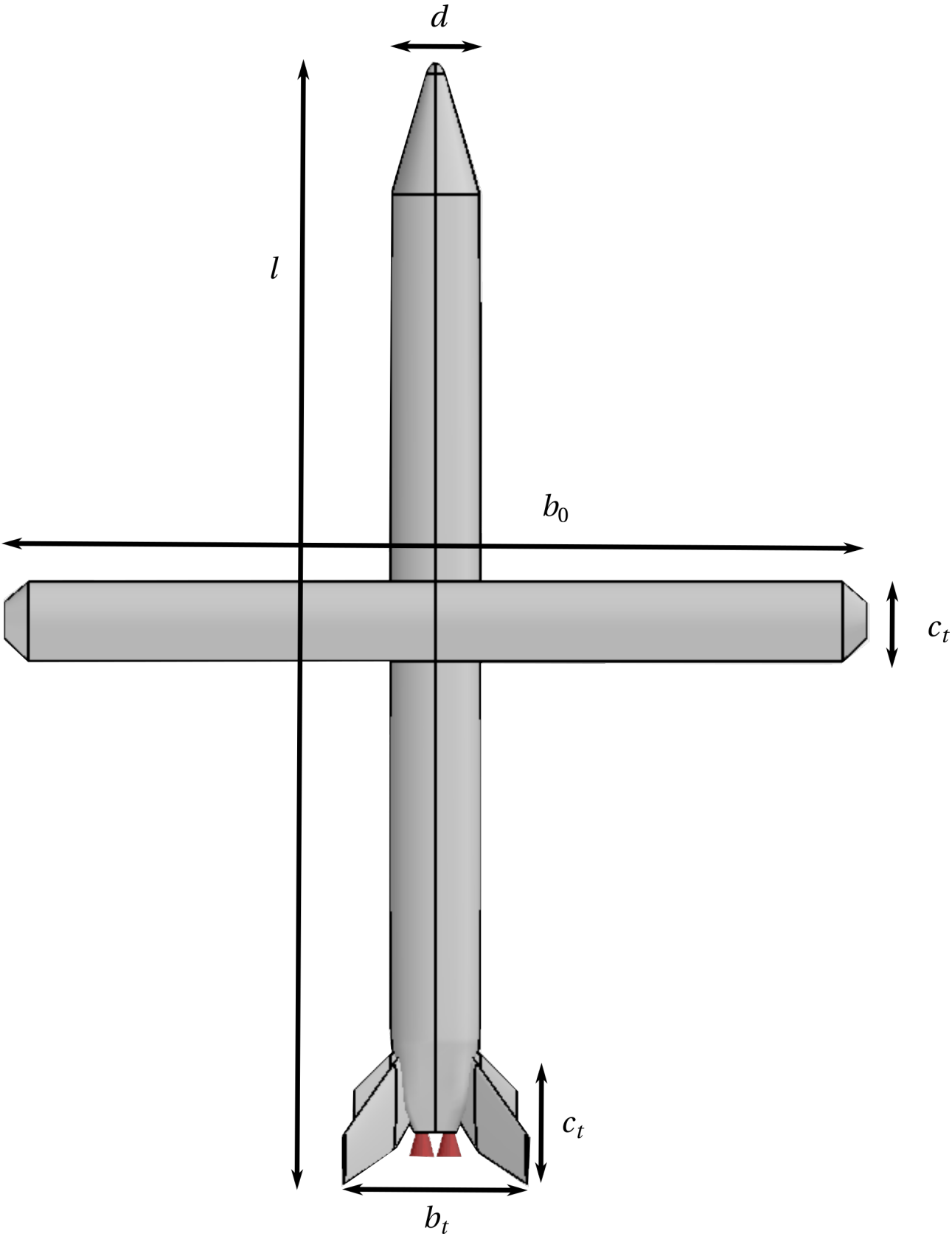


Figure A.1: Top view of the first stage booster with the wing deployed.

l	9.72 m
d	0.80 m
b_0	7.84 m
c_0	0.728 m
t	0.117 m
b_t	1.77 m
c_t	0.80 m
S	5.707 m

Table A.1: Booster geometry.

The mass properties of the first stage booster during the flyback are shown in Table **??**.

m	532 kg
m_w	59 kg
m_f	473 kg

Table A.2: Booster geometry.

The inertia tensor of the fuselage is

$$\mathbf{I}_f = \begin{bmatrix} 38.72 & 0.0 & -10.20 \\ 0.0 & 3961.5 & -10.0 \\ -10.2 & -10.0 & 3961.5 \end{bmatrix} kg * m^2, \tag{A.0.1}$$

and the inertia tensor of the wing at zero sweep is

$$\mathbf{I}_w = \begin{bmatrix} 302.4 & 0.0 & -0.1829 \\ 0.0 & 3.134 & 0.0 \\ -0.1829 & 0.0 & 3.134 \end{bmatrix} kg * m^2 \tag{A.0.2}$$

Appendix B

Lateral-Directional Rocket Controller Design

To investigate the effect of introducing elevons on the empennage to increase controllability, a lateral-directional controller is developed for the initial first stage booster design with and without the elevons. The lateral-directional state is

$$\mathbf{x}_{lat} = \begin{bmatrix} \beta & p & r & \phi \end{bmatrix}^T, \tag{B.0.1}$$

where β is the sideslip angle, p is the roll rate, r is the yaw rate and ϕ is the roll angle. The lateral-directional dynamics are represented in state space form as

$$\dot{\mathbf{x}}_{lat} = \mathbf{A}_{lat}\mathbf{x}_{lat} + \mathbf{B}_{lat}\mathbf{u}_{lat}. \tag{B.0.2}$$

The objective of the lateral-directional controller is to maintain a desired roll angle that is provided by guidance controllers not designed in this work. For this excercise, we assume that the controllers must maintain a constant zero roll angle. Atmospheric winds are the primary disturbance that must be rejected, which primarily results in sideslip angle perturbations.

B.1 Original Design: V-tail

The state matrix for the original design without the elevons is

$$\mathbf{A}_{lat} = \begin{bmatrix} -0.3333 & 0.0979 & -0.9846 & 0.0404 \\ -635.8 & -3.652 & 24.18 & 0.0 \\ 4.670 & 0.1360 & -1.275 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \end{bmatrix} \tag{B.1.1}$$

and the input matrix is

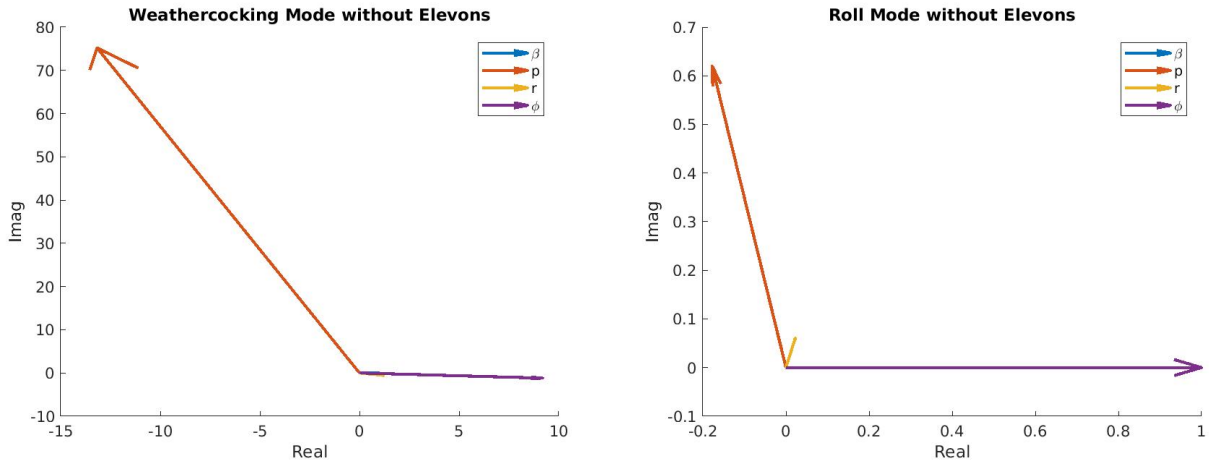
$$\mathbf{B}_{lat} = \mathbf{B}_{\delta_r} = \begin{bmatrix} 0.1775 \\ 510.1 \\ -22.54 \\ 0.0 \end{bmatrix} \tag{B.1.2}$$

with the input $\mathbf{u}_{lat} = \delta_r$. There are two oscillatory mode of motion, with the poles, damping and natural frequency shown in Table B.1. The fast mode is the weathercocking mode that describes the tendency of the booster to align with the flow, while the slow mode is a roll mode that returns the booster to wings level flight given a roll angle disturbance.

	Poles	Damping	Natural Frequency (rad/s)
Open loop without elevons	$p_{1,2} = -0.178 \pm 0.620i$ $p_{3,4} = -2.45 \pm 7.86i$	$\zeta_{1,2} = 0.275$ $\zeta_{3,4} = 0.298$	$\omega_{1,2} = 0.644$ $\omega_{3,4} = 8.22$

Table B.1: Poles for the open-loop dynamics without elevons.

The phasor diagrams for the two modes are shown in Figure B.1.



(a) (b)

Figure B.1: Phasor diagrams for open-loop modes without elevons: (a) weathercocking mode (b) roll mode.

The weathercocking mode phasor diagram is normalised by the sideslip angle, and shows that a small sideslip perturbation results in a significant roll angle and roll rate. The roll mode phasor diagram is normalisedby the roll angle, and shows that this mode has very little affect on yaw rate and sideslip angle, and thus consists of purely roll motions.

B.1.1 Roll Rate Damper

To prevent the large roll angle resulting from sideslip, the roll damping is first increased by feeding back the roll rate. Figure B.2 shows the block diagram for the roll rate damper without elevons.

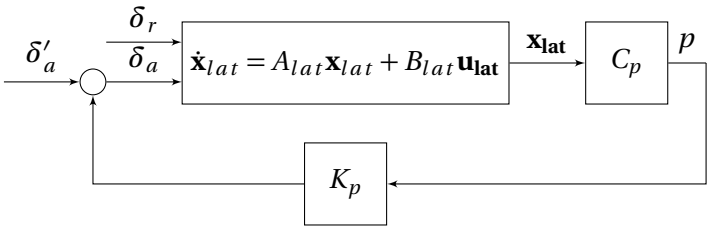


Figure B.2: Roll rate damper block diagram without elevons.

The output matrix for the roll rate is

$$\mathbf{C}_p = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}, \tag{B.1.3}$$

and the transfer function from rudder input to roll rate is obtained from the state space dynamics as

$$\frac{p(s)}{\delta_r(s)} = \mathbf{C}_p (s\mathbf{1} - \mathbf{A}_{lat})^{-1} \mathbf{B}_{\delta_r}. \tag{B.1.4}$$

The root locus diagram is plotted in MATLAB and shown in Figure B.3.

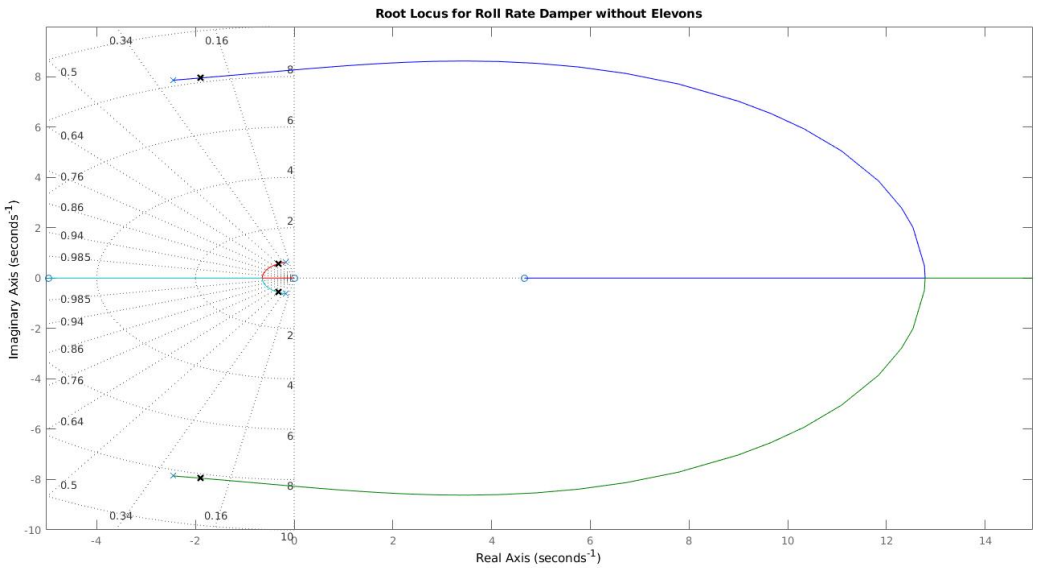


Figure B.3: Root locus design for roll-rate damper inner loop without elevons.

We see that while feeding back the roll improves damping of the roll mode, it begins to drive the weathercocking mode unstable. Thus a small gain of $K_p = -0.0016$ is chosen, with the resulting closed loop poles shown in Figure B.3. The closed loop dynamics are

$$\dot{\mathbf{x}}_{lat} = \left(\mathbf{A}_{lat} - \mathbf{B}_{lat} K_p \mathbf{C}_p\right)\mathbf{x}_{lat} + \mathbf{B}_{lat}\delta_r' \tag{B.1.5}$$

$$= \mathbf{A}_{RRD}\mathbf{x}_{lat} + \mathbf{B}_{lat}\delta_r' \tag{B.1.6}$$

B.1.2 Yaw Rate Damper

Next, the yaw rate is fedback to regain some of the damping of the weathercocking mode. The block diagram for the yaw rate damper is shown in Figure B.4.

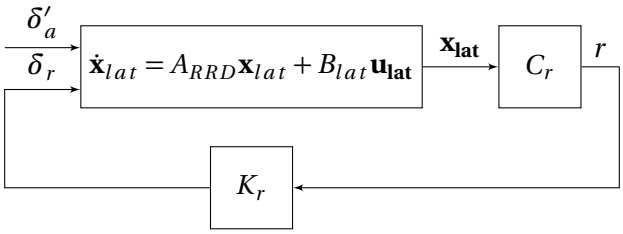


Figure B.4: Yaw rate damper block diagram without elevons.

The output matrix for the yaw rate is

$$\mathbf{C}_r = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}, \tag{B.1.7}$$

and the transfer function from pseudo rudder input to yaw rate is

$$\frac{r(s)}{\delta_r'(s)} = \mathbf{C}_r (s\mathbf{1} - \mathbf{A}_{RRD})^{-1} \mathbf{B}_{\delta_r}. \tag{B.1.8}$$

The root locus diagram is plotted in MATLAB and shown in Figure B.5.

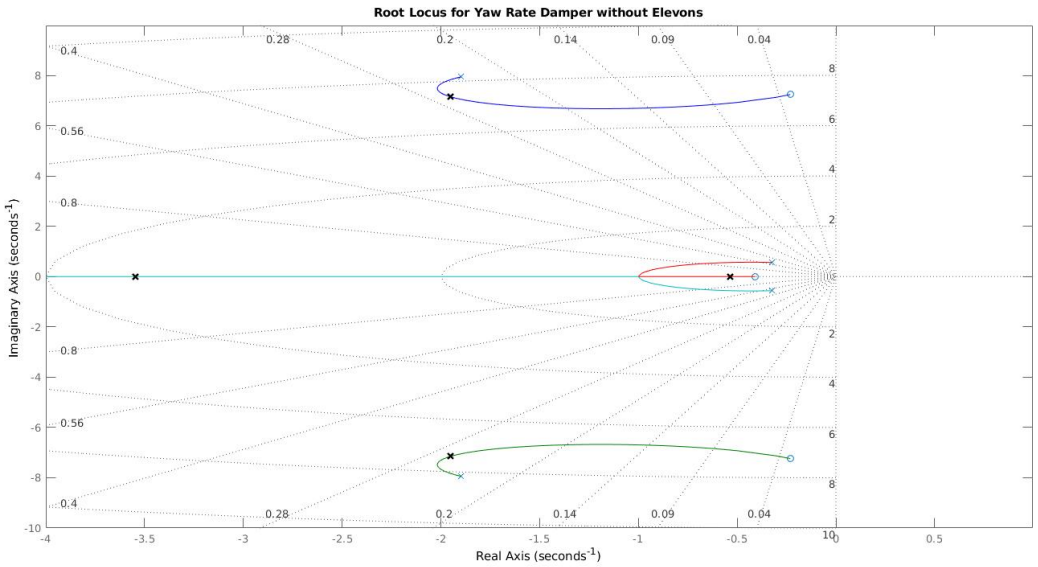


Figure B.5: Root locus design for yaw rate damper inner loop without elevons.

The gain $K_r = -0.157$ is chosen to increase the damping of the weathercocking mode as much as possible, and also helps to increase the damping of the roll mode. The closed loop dynamics with yaw rate feedback are

$$\dot{\mathbf{x}}_{lat} = (\mathbf{A}_{RRD} - \mathbf{B}_{lat} K_r \mathbf{C}_r)\mathbf{x}_{lat} + \mathbf{B}_{lat}\delta_r'' \tag{B.1.9}$$

$$= \mathbf{A}_{YRD}\mathbf{x}_{lat} + \mathbf{B}_{lat}\delta_r''. \tag{B.1.10}$$

B.1.3 Roll Angle Controller

Finally, an outer loop roll angle controller is designed, with a proportional-integral (PI) used to achieve zero steady state error. Figure B.6 shows the block diagram for the roll angle controller.

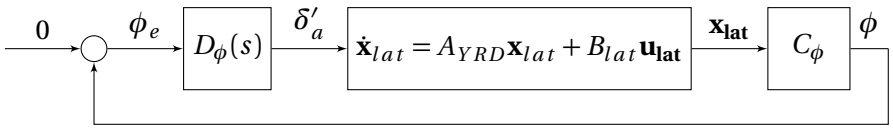


Figure B.6: Roll angle controller block diagram without elevons.

The roll angle output is

$$\mathbf{C}_\phi = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}, \tag{B.1.11}$$

and the transfer function is

$$\frac{\phi(s)}{\delta_r''(s)} = \mathbf{C}_\phi \left(s\mathbf{1} - \mathbf{A}_{YRD}\right)^{-1} \mathbf{B}_{\delta_r}. \tag{B.1.12}$$

The transfer function of the PI controller is

$$\frac{\delta_r''(s)}{\phi_{err}(s)} = K_\phi \frac{s+z}{s} = K_{p,\phi} + \frac{K_{i,\phi}}{s}. \tag{B.1.13}$$

The roll angle controller open loop transfer function is thus

$$\frac{\delta_r''(s)}{\phi_{err}(s)} \frac{\phi(s)}{\delta_r''(s)} = K_\phi \frac{s+z}{s} \mathbf{C}_\phi \left(s\mathbf{1} - \mathbf{A}_{YRD}\right)^{-1} \mathbf{B}_{\delta_r}. \tag{B.1.14}$$

The PI controller zero is chosen at -1, so $z = 1$ and the resulting root locus diagram is plotted in MATLAB and shown in Figure B.7.

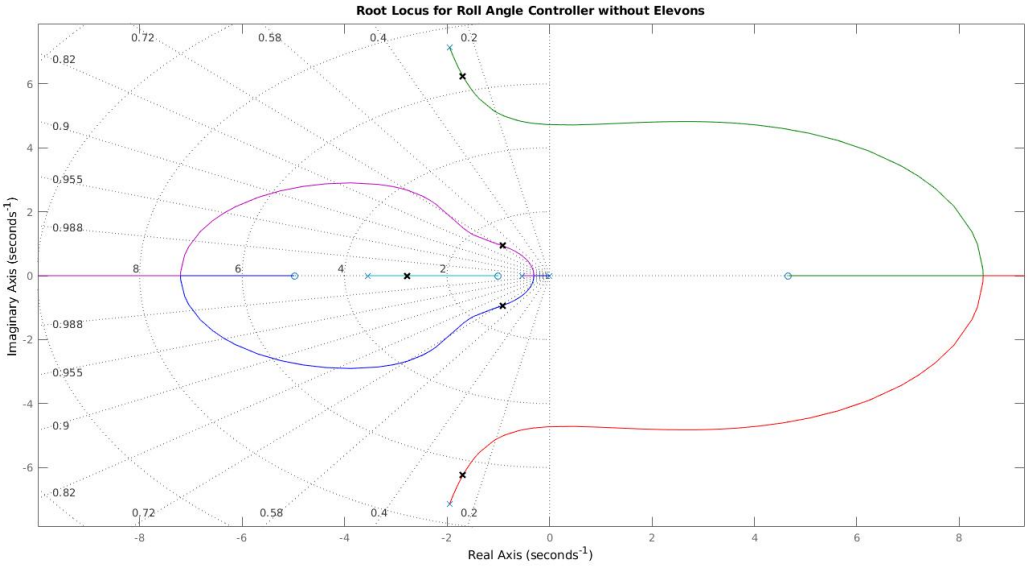


Figure B.7: Root locus design for roll angle controller without elevons.

Increasing the gain K_ϕ drives the weathercocking mode unstable, so the smallest gain achieving a critically damped roll mode is selected, giving $K_{p,\phi} = K_{i,\phi} = -0.0167$. With this controller, the closed loop dynamics poles, damping and natural frequency are given in Table B.2

	Poles	Damping	Natural Frequency (rad/s)
Closed loop without elevons	$p_{1,2} = -0.911 \pm 0.934i$ $p_{3,4} = -1.69 \pm 6.24i$ $p_5 = -2.78$	$\zeta_{1,2} = 0.698$ $\zeta_{3,4} = 0.261$	$\omega_{1,2} = 1.30$ $\omega_{3,4} = 6.47$

Table B.2: Poles for the open-loop dynamics without elevons.

B.2 Modified Design: V-tail and elevons

The state matrix for the modified design with the elevons is

$$\mathbf{A}_{lat} = \begin{bmatrix} -0.3421 & 0.0960 & -0.9846 & 0.0404 \\ -645.2 & -3.791 & 24.14 & 0.0 \\ 6.049 & 0.1187 & -1.319 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \end{bmatrix} \tag{B.2.1}$$

and the input matrix is

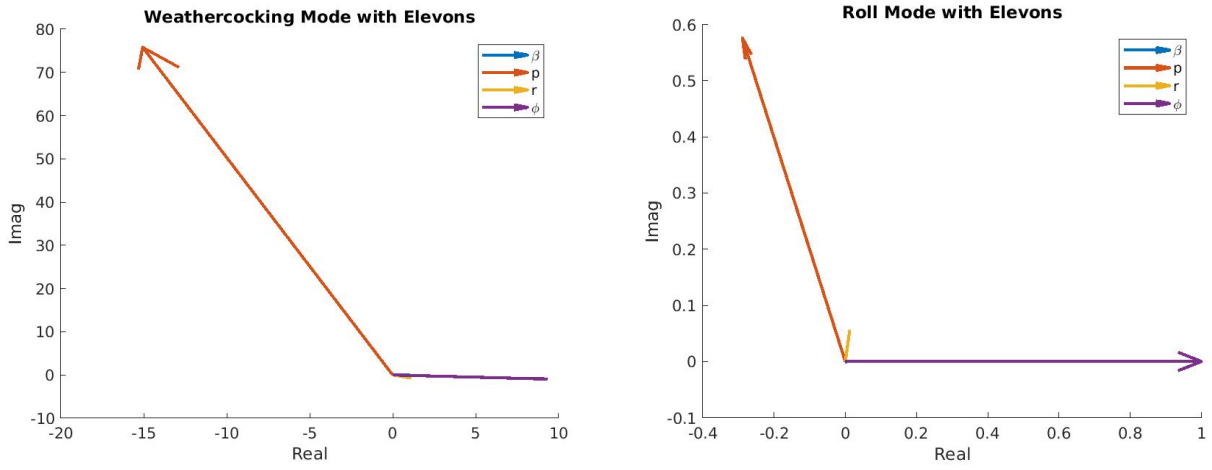
$$\mathbf{B}_{lat} = \begin{bmatrix} \mathbf{B}_{\delta_a} & \mathbf{B}_{\delta_r} \end{bmatrix} \mathbf{B}_{\delta_r} = \begin{bmatrix} 0.0489 & 0.1963 \\ -45.02 & 498.3 \\ -6.535 & -25.12 \\ 0.0 & 0.0 \end{bmatrix} \tag{B.2.2}$$

with the input $\mathbf{u}_{lat} = \begin{bmatrix} \delta_a & \delta_r \end{bmatrix}^T$. There are still two oscillatory mode of motion, with the poles, damping and natural frequency shown in Table B.3.

	Poles	Damping	Natural Frequency (rad/s)
Open loop with elevons	$p_{1,2} = -0.289 \pm 0.576i$ $p_{3,4} = -2.44 \pm 7.91i$	$\zeta_{1,2} = 0.448$ $\zeta_{3,4} = 0.294$	$\omega_{1,2} = 0.644$ $\omega_{3,4} = 8.28$

Table B.3: Poles for the open-loop dynamics with elevons.

The phasor diagrams for the two modes are shown in Figure B.8, and are very similar to the design without elevons.



(a) (b)

Figure B.8: Phasor diagrams for open-loop modes with elevons: (a) weathercocking mode (b) roll mode.

The same design process presented for the controller without elevons is followed to assess the per-

formance benefits provided by including the elevons. The primary difference is that now the elevons are used for the control input to dampen roll rates and to control the roll angle.

B.2.1 Roll Rate Damper

Figure B.9 shows the roll rate damper block diagram.

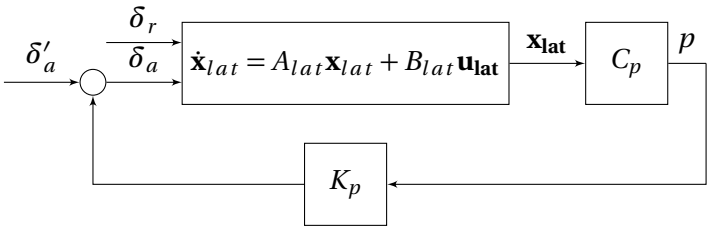


Figure B.9: Roll rate damper block diagram with elevons.

The output matrix for the roll rate is

$$\mathbf{C}_p = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}, \tag{B.2.3}$$

and the transfer function from elevon input to roll rate is obtained from the state space dynamics as

$$\frac{p(s)}{\delta_a(s)} = \mathbf{C}_p \left(s\mathbf{1} - \mathbf{A}_{lat}\right)^{-1} \mathbf{B}_{\delta_a}. \tag{B.2.4}$$

The root locus diagram is plotted in MATLAB and shown in Figure B.10.

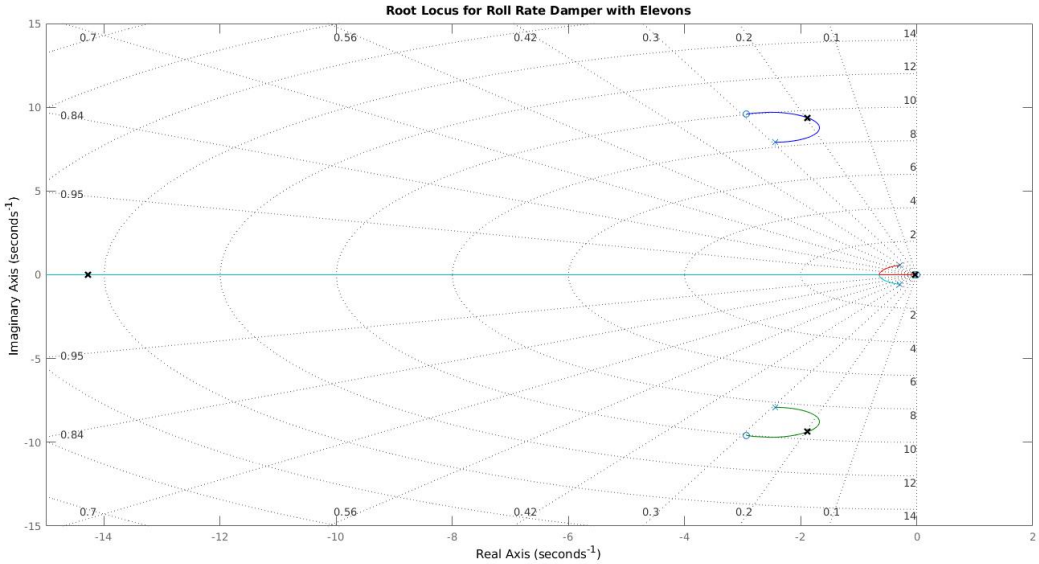


Figure B.10: Root locus design for roll-rate damper inner loop with elevons.

We immediately see a large difference as compared to using the ruddervator to feedback roll rate, as

the weathercocking mode now remains stable under feedback. Thus a large gain of $K_p = -0.28$ is chosen, with the resulting closed loop poles shown in Figure B.3. The closed loop dynamics are

$$\dot{\mathbf{x}}_{lat} = \left(\mathbf{A}_{lat} - \mathbf{B}_{\delta_a} K_p \mathbf{C}_p\right)\mathbf{x}_{lat} + \mathbf{B}_{lat} \begin{bmatrix} \delta'_a \\ \delta_r \end{bmatrix} \tag{B.2.5}$$

$$= \mathbf{A}_{RRD}\mathbf{x}_{lat} + \mathbf{B}_{lat} \begin{bmatrix} \delta'_a \\ \delta_r \end{bmatrix} \tag{B.2.6}$$

B.2.2 Yaw Rate Damper

Next, the yaw rate is fedback to increase the damping of the weathercocking mode. Figure B.11 shows the block diagram for the yaw rate damper.

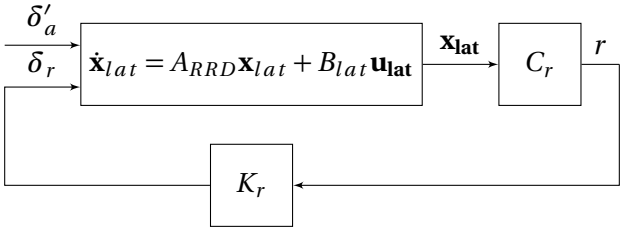


Figure B.11: Yaw rate damper block diagram with elevons.

The output matrix for the yaw rate is

$$\mathbf{C}_r = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}, \tag{B.2.7}$$

and the transfer function from rudder input to yaw rate is

$$\frac{r(s)}{\delta_r(s)} = \mathbf{C}_r \left(s\mathbf{1} - \mathbf{A}_{RRD}\right)^{-1} \mathbf{B}_{\delta_r}. \tag{B.2.8}$$

The root locus diagram is plotted in MATLAB and shown in Figure B.12.

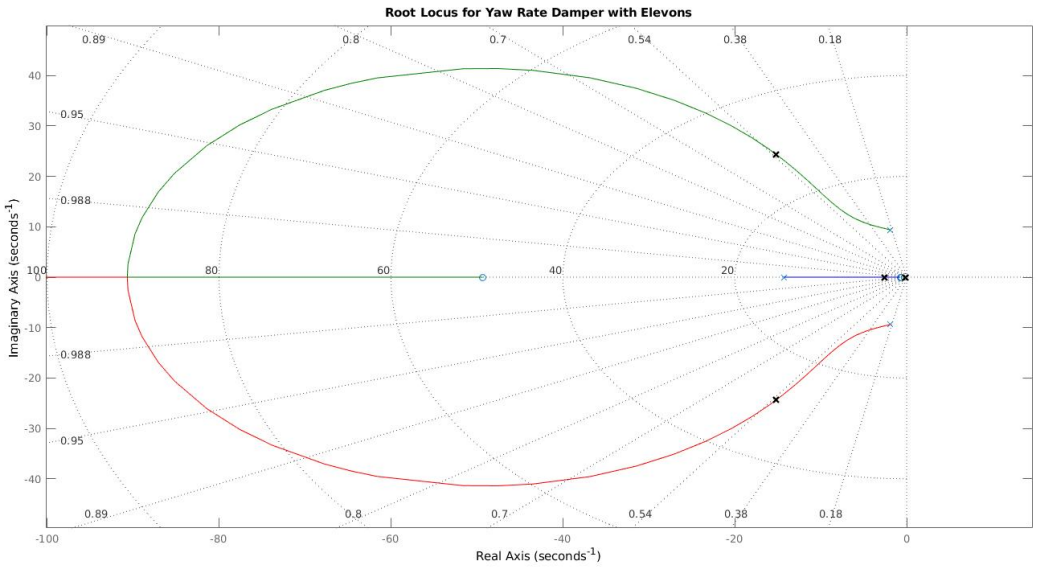


Figure B.12: Root locus design for yaw rate damper inner loop with elevons.

The gain $K_r = -0.6$ is chosen to increase the damping of the weathercocking mode as much as possible, and also helps to increase the damping of the roll mode. The closed loop dynamics with yaw rate feedback are

$$\dot{\mathbf{x}}_{lat} = \left(\mathbf{A}_{RRD} - \mathbf{B}_{\delta_r} K_r \mathbf{C}_r\right)\mathbf{x}_{lat} + \mathbf{B}_{lat} \begin{bmatrix} \delta'_a \\ \delta'_r \end{bmatrix} \tag{B.2.9}$$

$$= \mathbf{A}_{YRD}\mathbf{x}_{lat} + \mathbf{B}_{lat} \begin{bmatrix} \delta'_a \\ \delta'_r \end{bmatrix}. \tag{B.2.10}$$

B.2.3 Roll Angle Controller

Finally, an outer loop roll angle controller is designed using the elevon as the control input, with a proportional-integral (PI) used to achieve zero steady state error. Figure B.13 shows the block diagram for the roll angle controller.

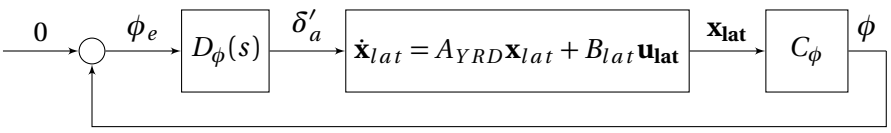


Figure B.13: Roll angle controller block diagram with elevons.

The roll angle output is

$$\mathbf{C}_\phi = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}, \tag{B.2.11}$$

and the transfer function is

$$\frac{\phi(s)}{\delta'_a(s)} = \mathbf{C}_\phi (s\mathbf{1} - \mathbf{A}_{YRD})^{-1} \mathbf{B}_{\delta_a}. \tag{B.2.12}$$

The transfer function of the PI controller is

$$\frac{\delta'_a(s)}{\phi_{err}(s)} = K_\phi \frac{s+z}{s} = K_{p,\phi} + \frac{K_{i,\phi}}{s}. \tag{B.2.13}$$

The roll angle controller open loop transfer function is thus

$$\frac{\delta'_a(s)}{\phi_{err}(s)} \frac{\phi(s)}{\delta'_r(s)} = K_\phi \frac{s+z}{s} \mathbf{C}_\phi (s\mathbf{1} - \mathbf{A}_{YRD})^{-1} \mathbf{B}_{\delta_a}. \tag{B.2.14}$$

The PI controller zero is chosen at -1, so $z = 1$ and the resulting root locus diagram is plotted in MATLAB and shown in Figure B.14.

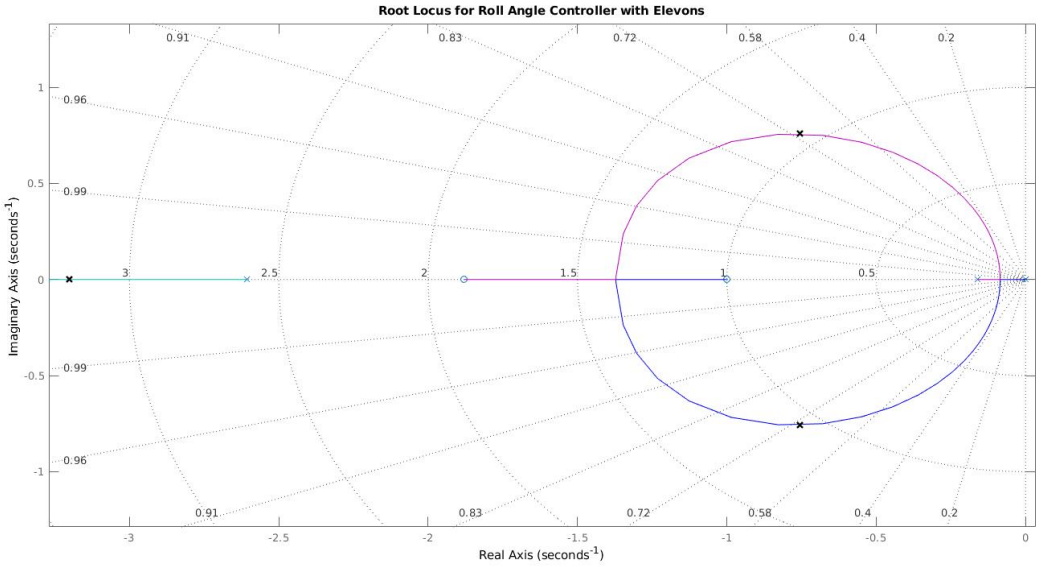


Figure B.14: Root locus design for roll angle controller with elevons.

Increasing the gain K_ϕ has a very small effect on the weathercocking mode, and the gain is selected to provided a critically damped roll mode as $K_{p,\phi} = K_{i,\phi} = -0.55$. With this controller, the closed loop dynamics poles, damping and natural frequency are given in Table B.4.

	Poles	Damping	Natural Frequency (rad/s)
Closed loop with elevons	$p_{1,2} = -0.755 \pm 0.758i$ $p_{3,4} = -14.2 \pm 24.3i$ $p_5 = -3.20$	$\zeta_{1,2} = 0.706$ $\zeta_{3,4} = 0.504$	$\omega_{1,2} = 1.07$ $\omega_{3,4} = 28.2$

Table B.4: Poles for the open-loop dynamics without elevons.

Appendix C

Linear Quadratic Servo Derivation

The cost function for the LQR tracker is

$$J = \frac{1}{2}\int_0^\infty [(C\mathbf{x}-\mathbf{r})^T \mathbf{Q}(C\mathbf{x}-\mathbf{r})+\mathbf{u}^T \mathbf{R}\mathbf{u}]d\,t \tag{C.0.1}$$

The results of the following derivation is given in [46] with the derivation itself left as an exercise. The derivation is shown here for completeness. The Hamiltonian of the system is

$$H = \frac{1}{2}[(\mathbf{C}\mathbf{x}-\mathbf{r})^T Q(\mathbf{C}\mathbf{x}-\mathbf{r})+\mathbf{u}^T \mathbf{R}\mathbf{u}]+\boldsymbol{\lambda}^T(\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{u}) \tag{C.0.2}$$

where $\boldsymbol{\lambda} \in \Re^8$ is an unknown multiplier. The state equation is given as

$$\dot{\mathbf{x}} = \frac{\partial H}{\partial \boldsymbol{\lambda}} = \mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{u} \tag{C.0.3}$$

The costate equation is

$$-\dot{\boldsymbol{\lambda}} = \frac{\partial H}{\partial \mathbf{x}} = \mathbf{C}^T\mathbf{Q}\,\mathbf{C}\mathbf{x}-\mathbf{C}^T\mathbf{Q}\mathbf{r}+\mathbf{A}^T\boldsymbol{\lambda} \tag{C.0.4}$$

The stationary equation is

$$0 = \frac{\partial H}{\partial \mathbf{u}} = \mathbf{R}\mathbf{u}+\mathbf{B}^T\boldsymbol{\lambda} \tag{C.0.5}$$

Solving Equation C.0.5 gives the control input

$$\mathbf{u} = -\mathbf{R}^{-1}\mathbf{B}^T\boldsymbol{\lambda} \tag{C.0.6}$$

In order to solve for $\boldsymbol{\lambda}$, the Hamiltonian system is setup as

$$\begin{bmatrix}\dot{\mathbf{x}}\\ \dot{\boldsymbol{\lambda}}\end{bmatrix} = \begin{bmatrix}\mathbf{A} & -\mathbf{B}\,\mathbf{R}^{-1}\mathbf{B}^T\\ -\mathbf{C}^T\mathbf{Q}\,\mathbf{C} & -\mathbf{A}^T\end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \boldsymbol{\lambda}\end{bmatrix} + \begin{bmatrix}0\\ \mathbf{C}^T\mathbf{Q}\end{bmatrix}\mathbf{r} \tag{C.0.7}$$

Since Equation C.0.7 is non-homogoneous, the Lagrangian multiplier $\boldsymbol{\lambda}$ is assumed to have the form

$$\boldsymbol{\lambda} = \mathbf{S}\mathbf{x} - \mathbf{v} \tag{C.0.8}$$

where $\mathbf{S}$ and $\mathbf{v}$ are unknowns to be determined. Thus the control input is

$$\mathbf{u} = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\mathbf{x} + \mathbf{R}^{-1}\mathbf{B}^T\mathbf{v} \tag{C.0.9}$$

with the Kalman gain $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}$. Taking the derivative of Equation C.0.8 results in

$$\dot{\boldsymbol{\lambda}} = \dot{\mathbf{S}}\mathbf{x} + \mathbf{S}\dot{\mathbf{x}} - \dot{\mathbf{v}} \tag{C.0.10}$$

Substituting in the state equation gives

$$\begin{aligned} \dot{\boldsymbol{\lambda}} &= \dot{\mathbf{S}}\mathbf{x} + \mathbf{S}(\mathbf{A}\mathbf{x} - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\boldsymbol{\lambda}) - \dot{\mathbf{v}} \\ &= (\dot{\mathbf{S}} + \mathbf{S}\mathbf{A} - \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S})\mathbf{x} + (\mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{v} - \dot{\mathbf{v}}) \end{aligned} \tag{C.0.11}$$

Now substituting in the costate equation results in

$$-\mathbf{C}^T\mathbf{Q}\mathbf{C}\mathbf{x} - \mathbf{A}^T\boldsymbol{\lambda} + \mathbf{C}^T\mathbf{Q}\mathbf{r} = (\dot{\mathbf{S}} + \mathbf{S}\mathbf{A} - \mathbf{S}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S})\mathbf{x} + (\mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{v} - \dot{\mathbf{v}}) \tag{C.0.12}$$

Rearranging and simplifying gives

$$0 = (\dot{\mathbf{S}} + \mathbf{S}\mathbf{A} + \mathbf{A}^T\mathbf{S} - \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} + \mathbf{C}^T\mathbf{Q}\mathbf{C})\mathbf{x} + (-\mathbf{A}^T + \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{v} - \mathbf{C}^T\mathbf{Q}\mathbf{r} - \dot{\mathbf{v}}) \tag{C.0.13}$$

In order for Equation C.0.13 to hold for all $\mathbf{x}$, both the first term and second term must equal zero. Setting the first term to zero gives

$$-\dot{\mathbf{S}} = \mathbf{S}\mathbf{A} + \mathbf{A}^T\mathbf{S} - \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S} + \mathbf{C}^T\mathbf{Q}\mathbf{C} \tag{C.0.14}$$

which is the differential Riccati equation. Taking the steady state value (i.e. $\dot{\mathbf{S}} = 0$), corresponding to the suboptimal solution to the performance index, Equation C.0.14 becomes the algabraic Ricatti equation, which is solved to give the Kalman gain $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}$. Substituting in the Kalman gain into the second term in Equation C.0.13 results in

$$-\dot{\mathbf{v}} = (\mathbf{A} - \mathbf{B}\mathbf{K})^T\mathbf{v} + \mathbf{C}^T\mathbf{Q}\mathbf{r} \tag{C.0.15}$$

Note that Equation C.0.15 is given in [46] and it is recommended that the complete time history be computed and stored online to aciheve optimal tracking. To simplify the implementation and the an-

laysis for multiple loops, it is assumed that the steady state value will provide sufficient performance. Suboptimal tracking is therefore achieved by using the steady state value of Equation C.0.15, providing

$$\mathbf{v} = -(\mathbf{A} - \mathbf{B}\,\mathbf{K})^{-T}\,\mathbf{C}^{T}\,\mathbf{Q}\mathbf{r} \tag{C.0.16}$$

Substituting Equation C.0.16 into Equation C.0.9 gives the control input as a function of the state and reference input

$$\begin{aligned}\mathbf{u} &= -\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{S}\mathbf{x} - \mathbf{R}^{-1}\mathbf{B}^{T}(\mathbf{A} - \mathbf{B}\,\mathbf{K})^{-T}\,\mathbf{C}^{T}\,\mathbf{Q}\mathbf{r} \\ &= -\mathbf{K}\mathbf{x} - \mathbf{L}\mathbf{r}\end{aligned} \tag{C.0.17}$$

This results in a controller that contains a feedback gain $\mathbf{K}$ to regulate the states and a feedforward gain $\mathbf{L}$ to track the trajectory. The feedforward gain also considers the effect of the feedback gain on the closed loop dynamics through the term $\mathbf{A} - \mathbf{B}\,\mathbf{K}$.